“PROPOSAL CLEARANCE SYSTEM”

Final Report

Submitted By

MANAN N. GANDHI (080280107012)

HIREN B. PANDYA (080280107023)

MEHUL B. SOJITRA (080280107043)

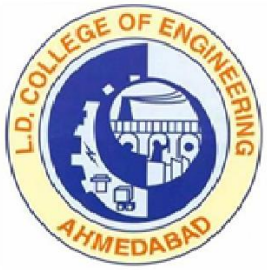
In fulfillment for the award of the degree

Of

BACHELOR OF ENGINEERING

in

Computer Engineering



L.D. College of Engineering, Ahmedabad

Gujarat Technological University, Ahmedabad

May, 2012

L.D. COLLEGE OF ENGINEERING

COMPUTER DEPARTMENT

2012

CERTIFICATE

Date: 23th May,2012

This is to certify that the project entitled 'Proposal Clearance System' has been carried out by MANAN N. GANDHI under my guidance in fulfilment of the degree of Bachelor of Engineering in COMPUTER ENGINEERING (7th Semester/8th Semester) of Gujarat Technological University, Ahmedabad during the academic year 2011-12.

Guide:

Kajal S. Patel
(Assistant Professor)
Computer Department
L.D. College of Engineering

Dr.(Prof.) J.S. Shah
Head of the Department
Computer Department
L.D. College of Engineering

L.D. COLLEGE OF ENGINEERING

COMPUTER DEPARTMENT

2012

CERTIFICATE

Date: 23th May,2012

This is to certify that the project entitled 'Proposal Clearance System' has been carried out by HIREN PANDYA under my guidance in fulfilment of the degree of Bachelor of Engineering in COMPUTER ENGINEERING (7th Semester/8th Semester) of Gujarat Technological University, Ahmedabad during the academic year 2011-12.

Guide:

Chirag A. Patel
(Assistant Professor)
Computer Department
L.D. College of Engineering

Dr.(Prof.) J.S. Shah
Head of the Department
Computer Department
L.D. College of Engineering

.

L.D. COLLEGE OF ENGINEERING

COMPUTER DEPARTMENT

2012

CERTIFICATE

Date: 23th May,2012

This is to certify that the project entitled 'Proposal Clearance System' has been carried out by MEHUL SOJITRA under my guidance in fulfilment of the degree of Bachelor of Engineering in COMPUTER ENGINEERING (7th Semester/8th Semester) of Gujarat Technological University, Ahmedabad during the academic year 2011-12.

Guide:

Mahesh Goyaniya
(Assistant Professor)
Computer Department
L.D. College of Engineering

Dr.(Prof.) J.S. Shah
Head of the Department
Computer Department
L.D. College of Engineering

ACKNOWLEDGEMENT

I owe great thanks to many people for supporting and guiding me in analysis of this project.

My deepest thanks to Assistant Professor Kajal S. Patel for guiding me throughout the analysis of the project and continuously assessing my work.

I express my deep sense of gratitude to Professor Chirag A. Patel for giving idea about the project and constantly motivating me in realising the feasibility and future of this project.

I am highly indebted to Assistant Professor Mahesh Goyani for giving valuable time in providing Technical guidance required for Proposal Clearance System.

Last but not the least I thank Gujarat Technological University for encouraging us to take a project and also my college and other staff members without whom this project would have been a distant reality.

ABSTRACT

This report presents a practical approach to a survey as one of requirements elicitation techniques. A fundamental goal in requirements engineering is to discover the employee/stakeholders' real needs. Conducting a good survey enables acquiring the appropriate information from the stakeholders or employee.

The report discusses the means to elicit software requirements through surveys and way to improve the given technique. Advantages of electronic surveys (e.g. Web- surveys) versus paper-surveys are pointed out. Architecture of Web-survey tool is proposed, taking the related work into the consideration. User-tool interactions are carefully analyzed. Solution for additional security and privacy issues is suggested.

Using data from Survey, we analyze how visual images embedded in a web-based survey can reduce nonresponse in the specific case that a respondent prematurely terminates the survey and preserve measurement validity. Page-by-page progression through the survey is modeled as a survival process with early termination seen as failure.

While images had no apparent effect on the termination process, respondent interest in the survey topic was linked to early termination. These results highlight the importance of placing interest-related questions early in the survey to better control for the effects of interest-driven attrition. Moreover, these findings suggest that an interactional information system approach, one that not only collects data but also pushes relevant information to respondents, may serve to generate or maintain interest and in the process reduce survey attrition.

LIST OF FIGURES

TABLE OF CONTENTS

1. INTRODUCTION

1.1 PROJECT SUMMARY

The Proposal Clearance system is to be developed to process documents electronically for improving productivity of organization and to track documents/proposals .

The application provides the secure environment for the data of Employee. Keep track of all transaction and detail information. It will easily generate invoices and reports. It enables a user to improve the accessibility, usability, security, approval, control of papers and electronic documents.

1.2 PURPOSE

Proposal Clearance System is a system where initiator is any user who creates documents and attaches files and workflow to the documents.

Processer can view all assigned document for respective actions in dashboard. He can process document and also change workflow runtime. Until the previous level processor submits the document, the next level processor is not able to view the document.

Currently, there is no system available to maintain the workflow in the organization. Employees use E-mails to forward or submit the task given to them. This is not efficient because employees are not aware that they got the mail. They have to be online all the time and they can't separate the forwarded task, rejected task and submitted task. So we are developing the system that solves all these problems.

1.3 SCOPE

In order to develop a reasonable project plan, it is required to functionally bind characteristics of the software so that by following the software Engineering principles.

The project management activity is the determination of software scopes. By considering the following objectives of software scopes, we derived the software scope statements. The

scope of this system is limited to the intranet in the company. The flow of data will be limited to Admin-Developer-Tester.

- Software scope must be ambiguous and understandable at management and technical levels.
- A statement of software scope must be bounded.
- There must be a quantitative data.

1.3.1 Context

The Proposal Clearance system is to be developed to process documents electronically for improving productivity of organization and to track documents/proposals .

1.3.2 Information Objectives

Manage Employees, Manage their Designation, Category details, Email, Login details, etc. are Admin visible data objects produced as output from Web Application. Document creation, Document category, Contents of Search, etc. are Employee level data objects produced as output from Web Application.

1.4 OBJECTIVES

- Provides account Management facility.
- Provides account generation facility.
- Employee can change the password and edit his/her profile.
- Can view Pending items, In Process items, Completed items, Action Pane in account.
- Provides Documentation Management facility
- Provides search facility.
- Search can be done based on category, report type and date.
- Search can be done based on all category or particular mentioned category.
- User can search files based on report type i.e. completed, archived, uncompleted, reference file.
- Files can only be viewed, they can't be modified.

- Admin can add and delete users and can also change the designation of the employee.

1.5 TECHNOLOGY AND LITERATURE OVERVIEW

1.5.1 Servlet and JSP

A **servlet** is a Java programming language class used to extend the capabilities of servers that host applications accessed via a request-response programming model. Although servlets can respond to any type of request, they are commonly used to extend the applications hosted by Web servers. Actually, it is a Java Applet that runs on a server instead of a browser.

A **Servlet** is a Java class in Java EE that conforms to the Java Servlet API, a protocol by which a Java class may respond to requests. They are not tied to a specific client-server protocol, but are most often used with the HTTP protocol. Therefore, the word "Servlet" is often used in the meaning of "HTTP Servlet". Thus, a software developer may use a servlet to add dynamic content to a Web server using the Java platform. The generated content is commonly HTML, but may be other data such as XML. Servlets are the Java counterpart to non-Java dynamic Web content technologies such as CGI and Servlets can maintain state in session variables across many server transactions by using HTTP cookies, or URL rewriting.

To deploy and run, the Apache Tomcat Server is used. It is an open source servlet container developed by the Apache Software Foundation (ASF). Tomcat implements the Java Servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, and provides a "pure Java" HTTP web server environment for Java code to run. The Servlet API, contained in the Java package hierarchy javax.servlet, defines the expected interactions of a Web container and a servlet. A Web container is essentially the component of a Web server that interacts with the servlets. The Web container is responsible for managing the lifecycle of servlets, mapping a URL to a particular servlet and ensuring that the URL requester has the correct access rights.

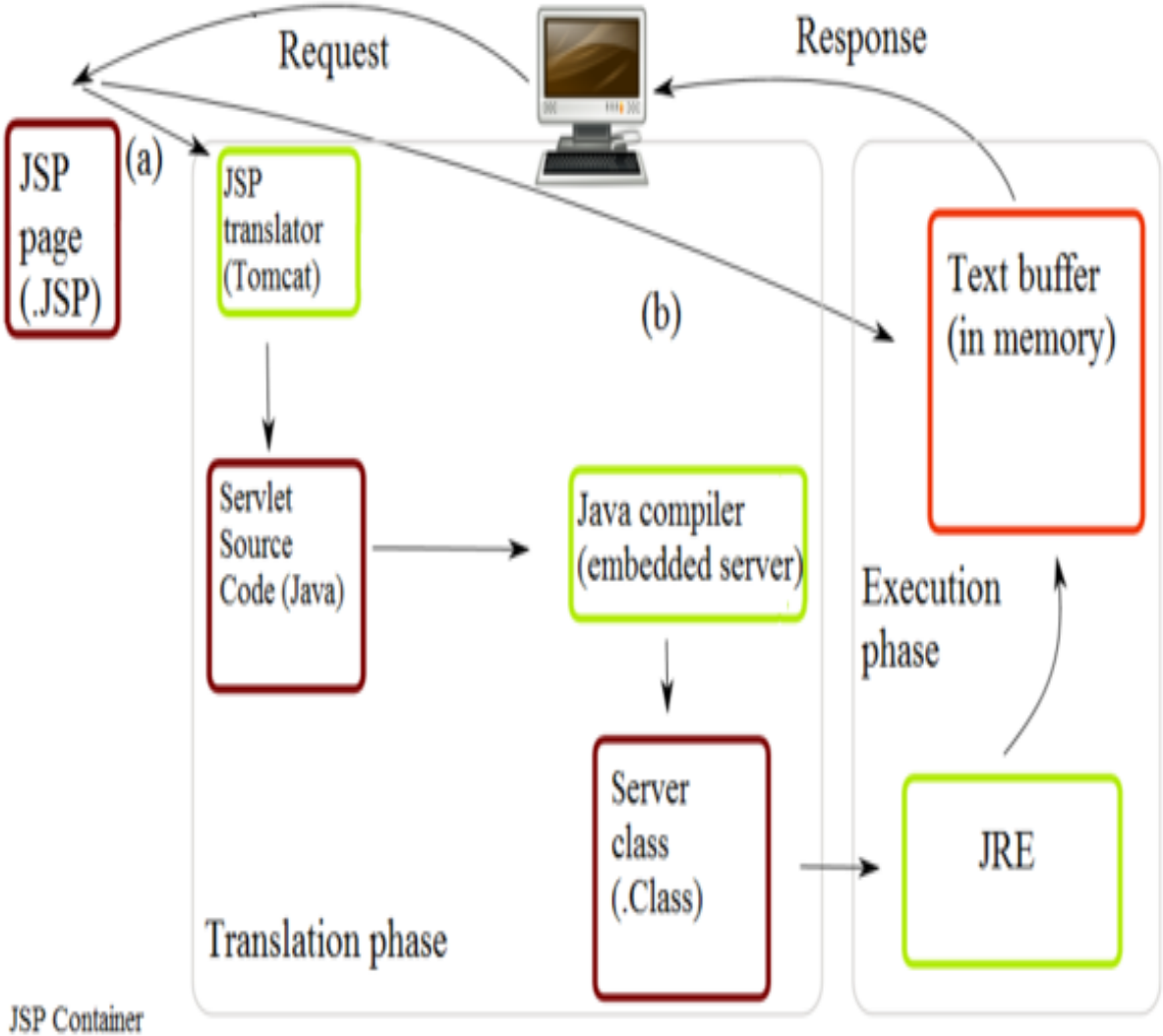


Fig: 1.2 JSP life cycle

2. PROJECT MANAGEMENT

2.1 PROJECT PLANNING

The objective of software project planning is to provide a framework that enables the manager to make reasonable estimates, cost, and scheduling. These estimates are made within a limited time frame at the beginning of a software project and should be updated regularly as the project progresses.

2.2 Project Development Approach and Justification

To solve actual problems in industry settings, software engineer or a team of engineers must incorporate development strategy that encompasses the process, methods and tools layers and generic phases. This strategy is often referred to as process model or an engineering paradigm. A process model for software engineering is chosen based on the nature of the project and application, the methods and tools to be used.

Spiral Model:

Justification

- In spiral model, the requirements are changing, so even if the requirements are given; they can also vary as per the new requirements of the user.
- The requirements in this project kept on changing as per the user modifications and hence,
- spiral model process flow was appropriate or such kind of a situation, and hence spiral model approach has been used.

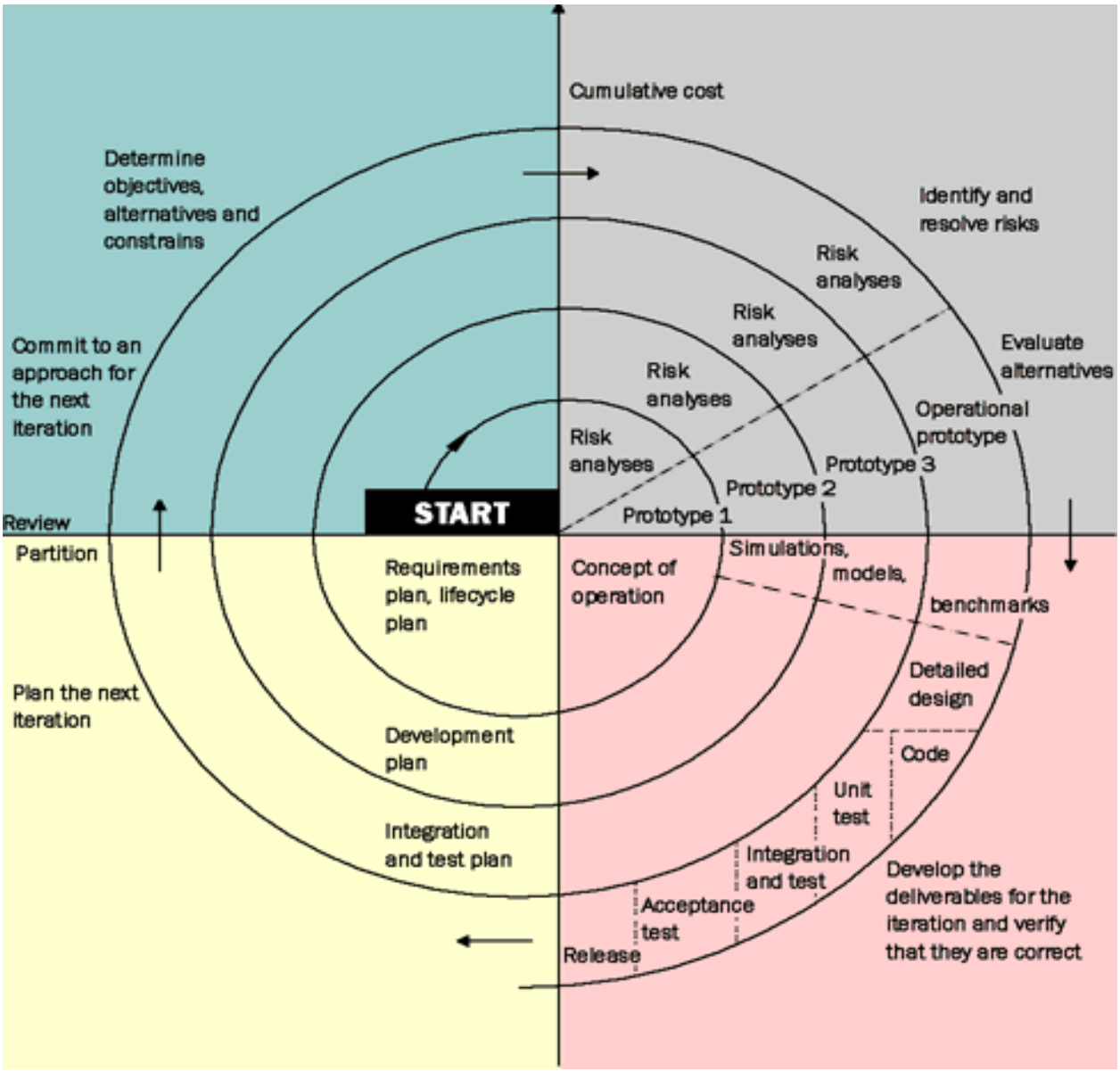


Figure 2.1 Spiral Model

- The risk assessment component of this model provides both developers and customers with a measuring tool that other models do not have. This model has great value of risk management.

- As the user's requirements were changing frequently we followed spiral model so that we can repeat the development process. And cope with the changing requirements. We have to verify each and every developed design or page with the user. And if changes were recommended again the development was carried out.

2.2.2 Software Scope

- The first activity in the project planning is the determination of software scope. Function and performance allocated to software during system engineering should be assessed to establish a project scope that is unambiguous and understandable at the management and technical levels.
- Software Scope described the data and control to be processed, function, performance, constraint, interface, and reliability. Defining the software scope for this project we have to collect information from the company networking person by asking some questions. These questions includes like from which server, how many clients, solutions to connect client to server etc.

2.2.3 Resources

The second software planning task is to estimation of the resources required to accomplish the software development effort. The above figure illustrates development resources as a pyramid. The development environment – hardware and software tools-sits at the foundation of the resources pyramid and provide the infrastructure to support the development effort. At a higher level, we encounter reusable software components-software building blocks that can dramatically reduce development costs and accelerate delivery. At the top of the pyramid is the primary resource-people.

2.2.3.1 Human Resource

The human resources required are Database, network administrator (Project guide).

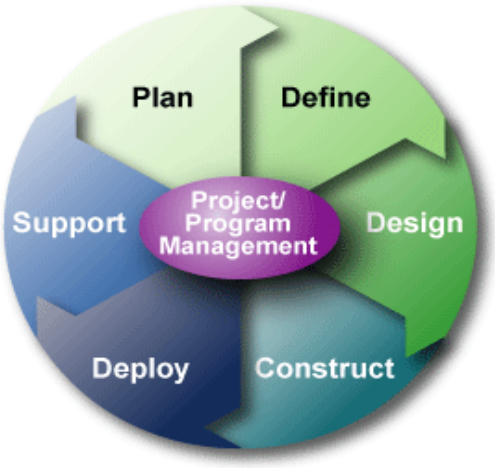
2.2.3.2 Reusable Software Resources

For our project there is no existing specification, design, code or data of past projects are included. But the new components are building for our need.

2.3 PROJECT SCHEDULING

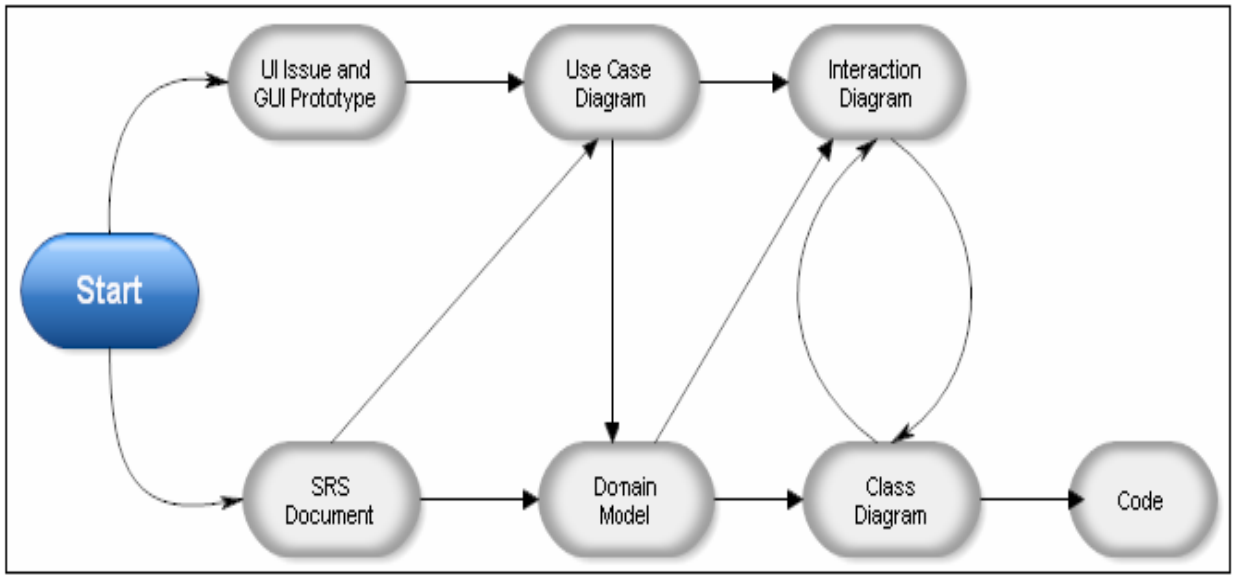
2.3.1 Basic Principal

"Software project scheduling is an activity that distributes estimated effort across the planned project duration by allocating the effort to specific software engineering tasks."



Proper scheduling requires:

- All tasks appear in network.
- Effort and timing are intelligently allocated to each task.
- Interdependencies between tasks are properly indicated.
- Resources are allocated for the work to be done.



2.3.3 TimeLine Chart

2.3.3.1 Time Allocation

Each task to be scheduled must be allocated some number of work units. In my software project, time allocated to task that check server is connected. The time that was spent in each part of system development is shown in the activity chart.

2.3.3.2 Task Sets

We have selected the "Spiral Model" so that there are six different work tasks to work together. A tasks set is a collection of software engineering work tasks, milestones, and deliverables that must be accomplished to complete a particular project. Tasks set are designed to accommodate different types of projects and different degree of Rigor. Most software organizations encounter the following projects.

- Concept Development.
- New Application development.
- Application enhancement.
- Application maintenance.
- Re-engineering project.

Refinement of major tasks

The above tasks are again refinement as follows:

- Analysis of required system
  - 1) Study of required flow
  - 2) Study of methodology
- Defining goals and objectives
  - 1) Preparing the goal
  - 2) Defining the flow of the project
- Finding out resources required
  - 1) Identifying the resources
  - 2) Making arrangement for getting the resources
- Coding and Testing.
  - 1) Designing the tables and creating them.
  - 2) Coding for the Forms.
  - 3) Validation of the forms.
  - 4) Testing
- Documentation
  - 1) Divide document in small parts
  - 2) Documentation of each part
  - 3) Integration of all parts
  - 4) Review of project documentation

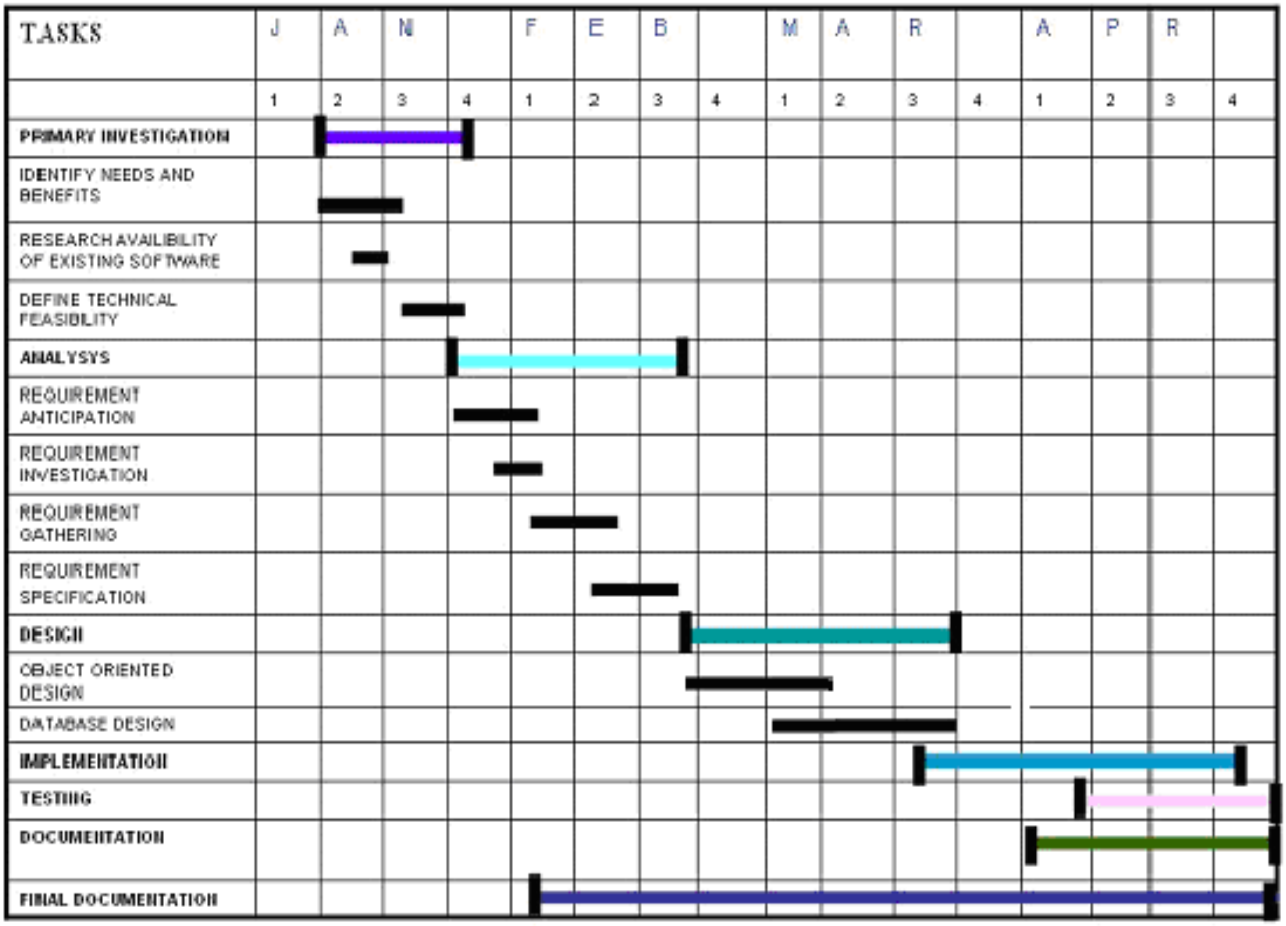


Figure 2.4 Time Line Chart

2.4 RISK MANAGEMENT

Software is a difficult undertaking. Lots of things can go wrong and frankly, many often do. It's for this reason that being prepared understanding the risks and taking proactive measure to avoid or manage them is a key element of good software project management. Recognizing what can go wrong is the first step called „Risk Identification". Next each risk is analyzed to determine the likelihood that it will occur and the damage that it will do if it does occur. Once this information is established, risks are ranked, by probability and impact. Finally a plan is developed to manage those risks with high probability.

2.4.1 Risk Identification

Risk identification is a systematic attempt to specify threats to the project plan. By identifying known and predictable risks, I take first step towards avoiding them when possible and controlling them when necessary.

2.4.2 Risk identification artifacts

I considered the following types of risk to identify the risk in proper manners. The next table shows the type of risks.

Table 2.1 Risk Type

Risk Type	Description
Project Risks	This type of risk can threaten the project plan. If project risks become real, then project schedule problems and their impact on a software project.
Technical Risks	This type of risk can threaten the quality and timeliness of software to be produced. If a technical risk becomes a reality then implementation may become impossible or difficult.
Business Risks	This type of risk can threaten the viability of the software to be built. Business risks often the project or the product.

2.4.3 Risk Projection

Risk projection ,also called risk estimation, attempts to rate each risk in two ways-the likelihood or probability that risk is real and consequences of the problems associated with the risk should it occur. The following table shows the artifacts used in the risk projections.

Risk category	**Description**
Catastrophic	Risk can cause the whole system to fail.
Critical	There may be significant degradation of the system may result
Minor	The risk can be easily recovered from the system failure.
Negligible	The risk can be negligible and shall not affect in the performance in the system at all.

2.4.4 Risk Refinement

I should always ready for an unforeseen event that may or may not occur. The strategy that I had used in for the risk refinement is risk management and contingency planning because it assumes that mitigation efforts have failed and that the risk has become a reality. The next table shows the risks planning and steps which shall be taken to overcome the risks

2.4.5 Tracking and Control

Weekly status reports were sent to the Teachers and each module discusses in the meetings. Information about work done and the further working report is exchanged during the weekly meetings to make sure is on schedule to meet all our milestones.

Table 2.3 Tracking and Control of Risk Types

Risk ID#	Risk Type	Impact	Affected component
1	Project/Catastrophic	Critical	Schedule
2	Business/Critical	Critical	Performance
3	Project/Minor	Critical	Performance
4	Technical/Minor	Critical	Performance
5	Technical/Critical	Minor	Performance
6	Technical/Catastrophic	Critical	Performance
7	Project/Minor	Minor	Support
8	Business/Critical	Critical	Schedule
9	Business/Critical	Critical	Support

3. SYSTEM REQUIREMENTS STUDY

3.1 User Characteristics

The users who are going to use this web application must have Login ID and Password provided by Administrator. The users have the knowledge of Intranet and the web application is made according to their convenience. Users of this System are as follows:

- Administrator
- Employee

3.2 Hardware and Software Requirement

3.2.1 Hardware Requirements:

- Processor: Dual Core processor
- RAM: 512MB
- Hard Disk: Necessary Space,3 GB installation drive
- Monitor: 14" color Monitor
- Mouse: mouse or compatible pointing device
- Keyboard: Any Keyboard

3.2.2 Software Requirements:

- OS: Windows 2000 or higher/server
- IDE : NetBeans IDE-6.8
- Technology : JSP, Servlet, Java mail API
- Web Technologies: JavaScript, AJAX, JSON
- Web Server: Tomcat 6.0
- Database(back end) : MYSQL
- Supportive Development Tools: Microsoft Words 2003/2007 , Microsoft Visio

3.3 Functional and Non functional Requirements

3.3.1 Functional Requirements

- **User Interfaces**

  In this system, there are four users which plays different roles.

  - Employee
  - Administrator

- **Employee Interface**

  - Login
  - Change password
  - Create Workflow
  - Create Document
  - Attach Document
  - View the Document report

- **Administrator Interface**

  - Login
  - Forgot password
  - Change password
  - Manage Employee profile
  - Manage business rules
  - Change the Designation of Employee
  - Add, edit and delete users

3.3.2 Non Functional Requirements

There are lots of features that we have included in the software. But some of them were described over here.

3.3.2.1 Easy to maintain records

All the records are stored in the site database. Back up can be taken easily.

3.3.2.2 Less Time Consuming

All the entries are done easily so it requires less time and also calculation is done automatically through computer so take less time.

3.3.2.3 Editing of Data

Easy to change the records.

3.3.2.4 No redundancy of data

If the admin is having the reservation the some of the information is automatically filled up at the time of registration. So redundancy of data decreases.

3.3.2.5 Security of data

Security of data is important factor of the software success. We are providing the security for the database so no one can illegal access to the database.

4 SYSTEM ANALYSIS

4.1 Study of Current System

The Proposal Clearance system is to be developed to process documents electronically for improving productivity of organization and to track documents/proposals . The application provides the secure environment for the data of Employee. Keep track of all transaction and detail information. It will easily generate invoices and reports. It enables a user to improve the accessibility, usability, security, approval, control of papers and electronic documents.

Proposal Clearance System is a system where initiator is any user who creates documents and attaches files and workflow to the documents. Processer can view all assigned document for respective actions in dashboard. He can process document and also change workflow runtime. Until the previous level processor submits the document, the next level processor is not able to view the document.

4.2 Requirements of new system

- Currently, there is no system available to maintain the workflow in the organization. Employees use E-mails to forward or submit the task given to them. This is not efficient because employees are not aware that they got the mail. They have to be online all the time and they can't separate the forwarded task, rejected task and submitted task. So we are developing the system that solves all these problems.

  The system mainly contains the feature:
- Id generation automatically accordingly grade of Employee
- Browser Compatibility
- Platform independence
- Data Security
- High Performance
- Improves efficiency and accuracy.
- Reduce manpower.
- User friendly.

- Flexible reporting

4.3 System Modules

- **Account Management**

.

This system enables different level employees to make an account. The account is managed as per the designation of the employee.

Login:

- Process to allow authorized user.
- For a new user, do registration.
- If employee is not remembered their current password then he further proceed (E.g. Security question).
- Can change the password by confirming the old password.

There are four sections in an account to see the documentations details:

- **Pending items:**

No	Name	Description
1	Received for Action	Click to see the Documents received for action
2	Received Back	Click to see the Documents received back from Higher Work Flow user.
3	Received as Cc	Click to see the Documents received as Cc.
4	Received Approved	Click to see the approved documents initiated by any user.
5	Draft	Click to see the draft documents which are Initiated by any user but pending for work flow attachment.

- **In Process items :**

No	Name	Description
1	Sent created	Click to see the list of documents which are sent for approval created by any user.
2	Sent Forward	Click to see list of documents which are sent forward by you for further action
3	Sent Back	Click to see the Documents which are sent back by you.
4	Sent Approved	Click to see the Documents which are approved and sent to originator.

- **Completed items:**

No	Name	Description
1 (1) (2) 2 (3) (4) (5)	Rcved for reference	Click to see the Documents which are approved and received for your reference.
	Archived	Click to see list of your documents which are approved and archived for future reference as a originator user.

- **Action Pane**

No	Name	Description
1	Search file	Click to search any document based on date, category and report type.
2	Change password	Click to change the existing password.
3	Edit profile	Click to edit your own user profile.
4	Help	Click to open Online help document.

➢ Document Management :

- Initiator is any user who has rights to create a document/proposal in the application.
- Document/Proposal has been divided into categories, based on the available list initiator has to enter document category. Initiator can also create its own favourite category list whereby including those category which is being used maximum by him.
- After creating file/Document, Workflow attachment process is not mandatory to finish at same time. If initiator leaves the workflow attachment process then document will be saved as draft document for further use. Initiator can select draft document to continue workflow attachment process at any time. Initiator can then proceed to upload attachments for the same. Initiator can attach archived file as well as reference file provided by the admin.
- Initiator selects users from the list available to create the approval workflow for the document created. By default, selected users will be treated as intermediate processors. Document will flow from one processor to another in the same order as they are included in the list.
- Processor can see list of pending documents from 'Rcvd for Action' tab from dashboard. He can process document for further action by clicking 'Process file' button and View detail by clicking 'View Detail' button.
- When processor open document for processing he can see document header detail, workflow detail, noting detail entered by previous level processor, attachment detail. Intermediate processors can either approve the document and send to the next processor in the workflow or can send the document back to the previous level processor.
- At last document goes to terminator, now when terminator approves document it goes to Initiator for archival. Once the document is archived initiator can only view archived document but cannot process it.

➢ Search Facility :

- User can search files based on category, report type and date.
- User can search based on all category or particular mentioned category.
- User can search files based on report type i.e. completed, archived, uncompleted, reference file.
- Files can only be viewed, they can't be modified.

➢ Admin :

- Manage Employee Profile.
- Maintain the business rules and system features.
- Can add, edit, delete users
- Can change the designation of employee
- Allow access to specific user

4.4 Feasibility Study:

Feasibility Study is the likelihood the system will be useful to the Organization. After studying the requirements, whether the proposed project is feasible or not, is determined by checking the various feasibilities. The three aspects in the feasibility study portion of preliminary investigation are:

1) Operational Feasibility:

- Operational feasibility focuses on whether the system will work when it is developed and installed. Operationally the system is feasible because Feasibility Analysis.
- The proposed System will not cause any harm to the existing system and its users
- No special training required for the user as it has self –explanatory interface.
- Validation of data input is taken care of by the system and not by the user.

2) Technical Feasibility:

- Technically the project is feasible because "necessary technology "exists to do what is suggested and required by the organization.
- The hardware needed to develop and implement the system is adequate. "**PROPOSAL CLEARANCE SYSTEM**" develops using JSP, Servlet, Java mail API, SQL-Server 2005 for development. The software guarantees accuracy, reliability and ease of access and data security.

3) Economic Feasibility:

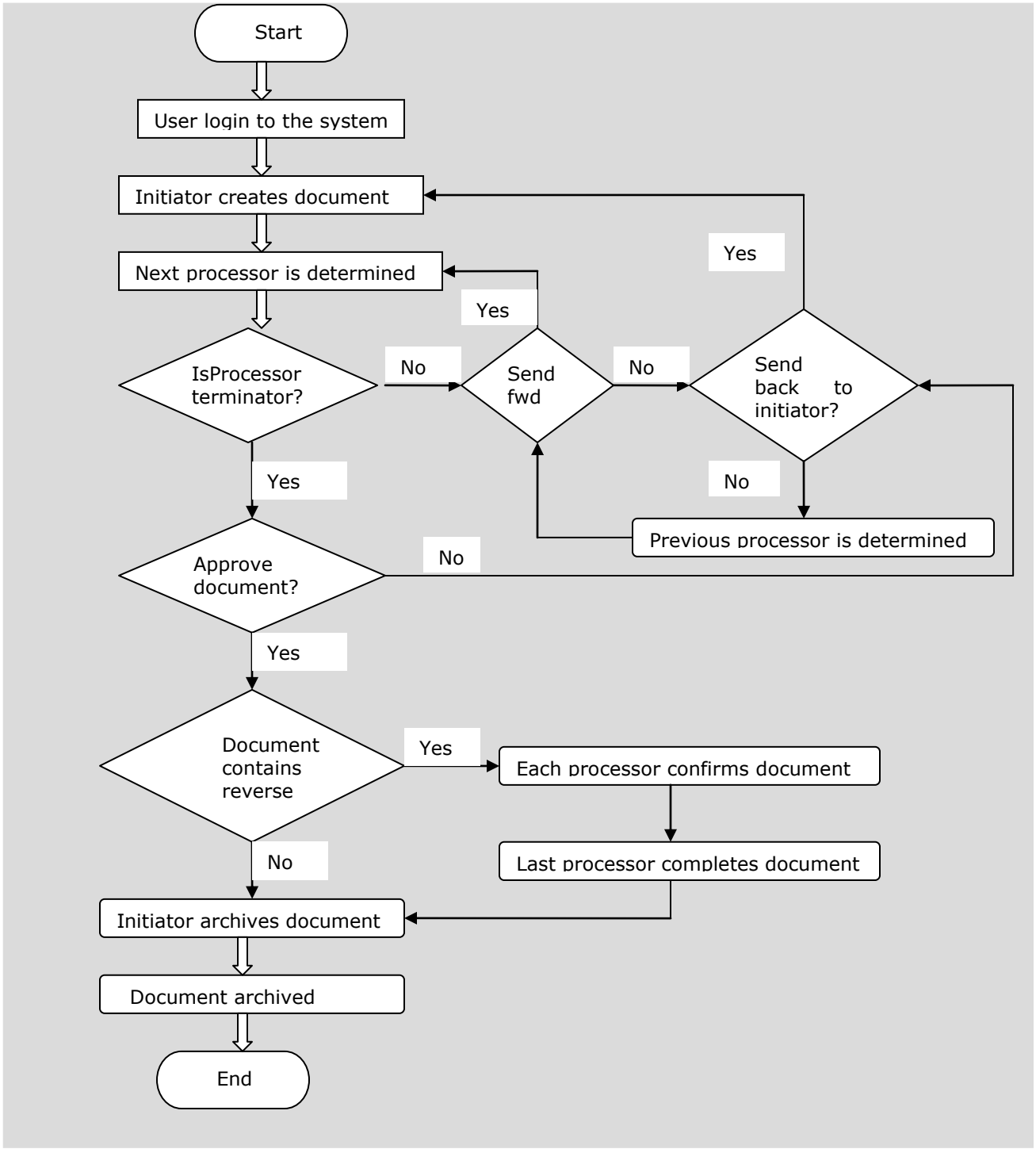
A system that can be developed and that will be used if installed must still be a good investment for the organization. Financial benefits must equal or exceed the costs. The financial and economic issues raised are as under:

- No extra cost is incurred for developing the system after the system is developed once. As required software are already used by the department.
- No extra cost for the modification or additional of software and hardware will require in case of future expansion of the current system.

So the proposed system is financially and economically feasible.

Diagrams:

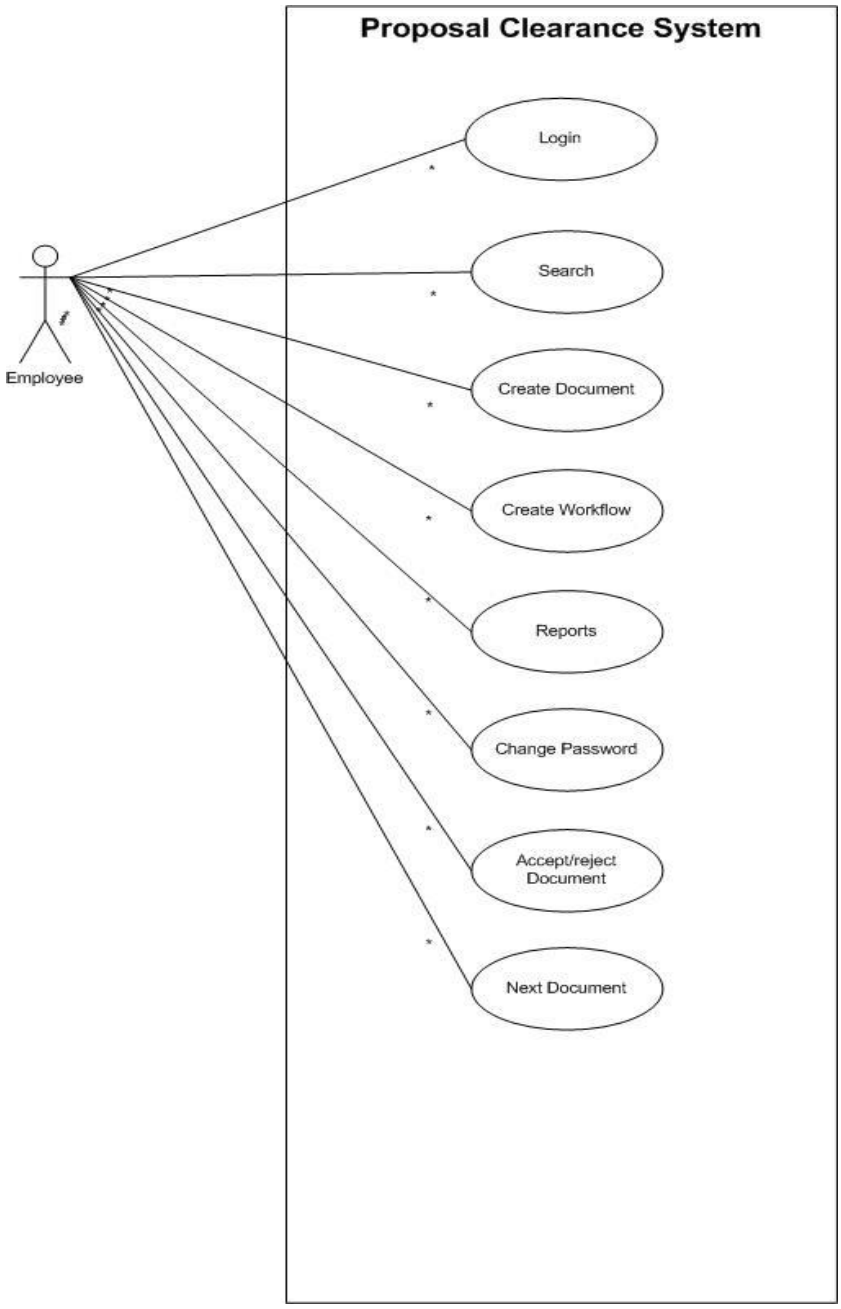
Process Flow:



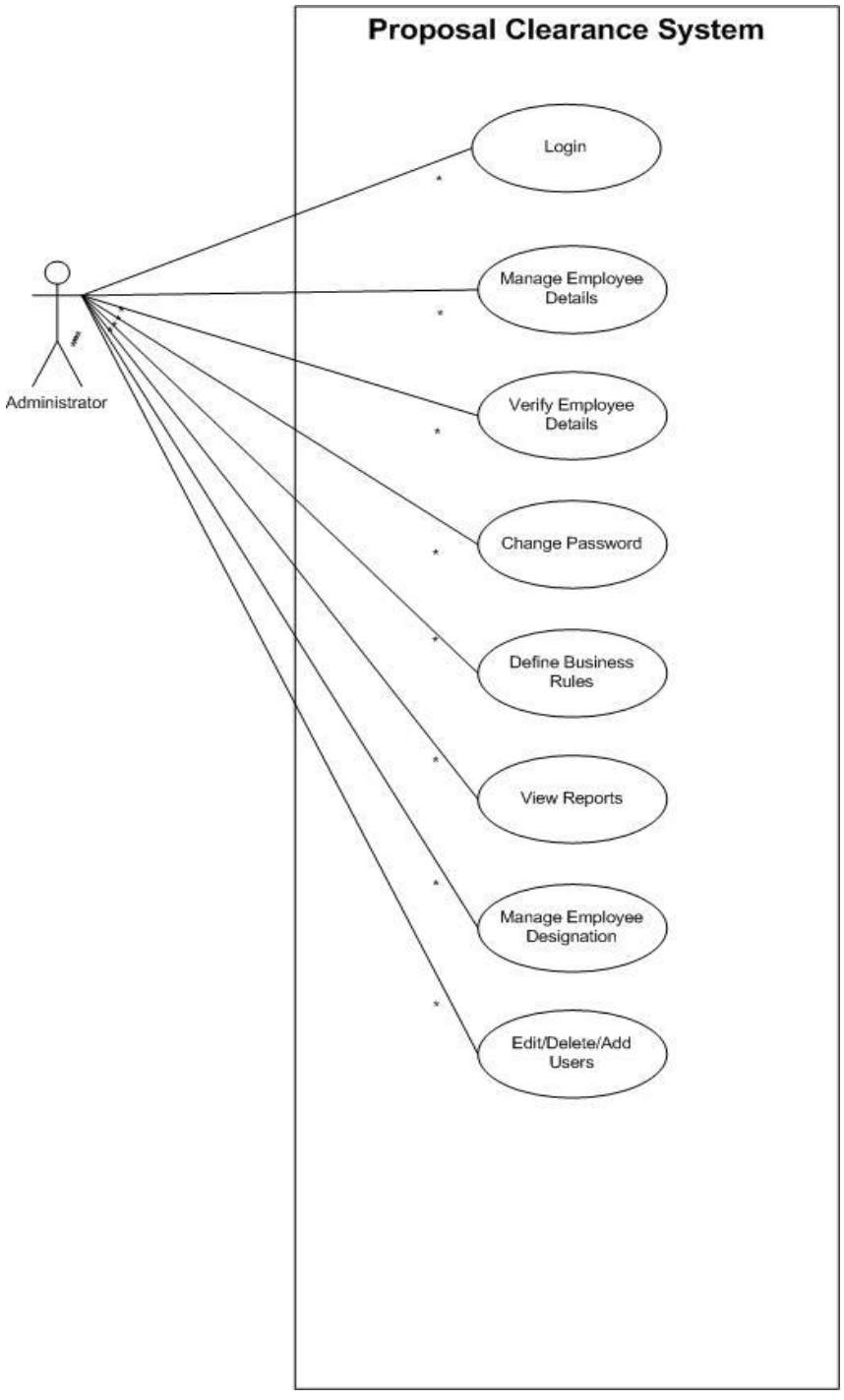
Use Case Diagram:

A **use case diagram** in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.Use Case diagrams are formally included in two modeling languages defined by the OMG: the Unified Modeling Language (UML) and the Systems Modeling Language (SysML).

Use Case Diagram of Employee :



Usecase Diagram for Administrator:



Sequence Diagram:

A **sequence diagram** in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams. A sequence diagram shows, as parallel vertical lines (*lifelines*), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner. For instance, the UML 1.x diagram on the right describes the sequences of messages of a (simple) restaurant system. This diagram represents a Patron ordering food and wine, drinking wine then eating the food, and finally paying for the food. The dotted lines extending downwards indicate the timeline. Time flows from top to bottom. The arrows represent messages (stimuli) from an actor or object to other objects. For example, the Patron sends message 'pay' to the Cashier. Half arrows indicate asynchronous method calls. The UML 2.0 Sequence Diagram supports similar notation to the UML 1.x Sequence Diagram with added support for modeling variations to the standard flow of events.

Diagram building blocks

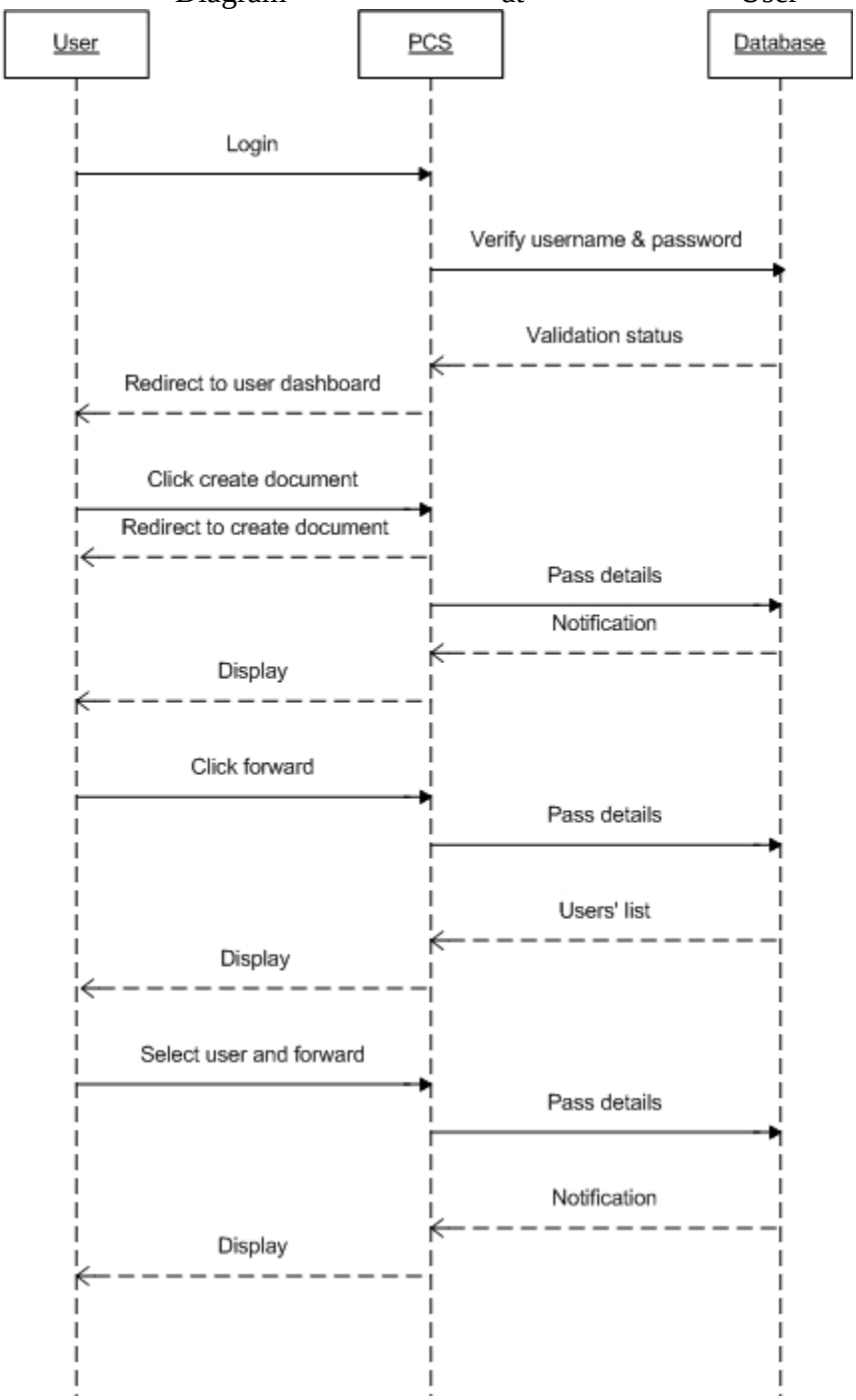
If the lifeline is that of an object, it demonstrates a role.Note that leaving the instance name blank can represent anonymous and unnamed instances.

In order to display interaction, messages are used. These are horizontal arrows with the message name written above them. Solid arrows with full PRJ/2011/IT/02 SYSTEM ANALYSIS CITC (IT) 46

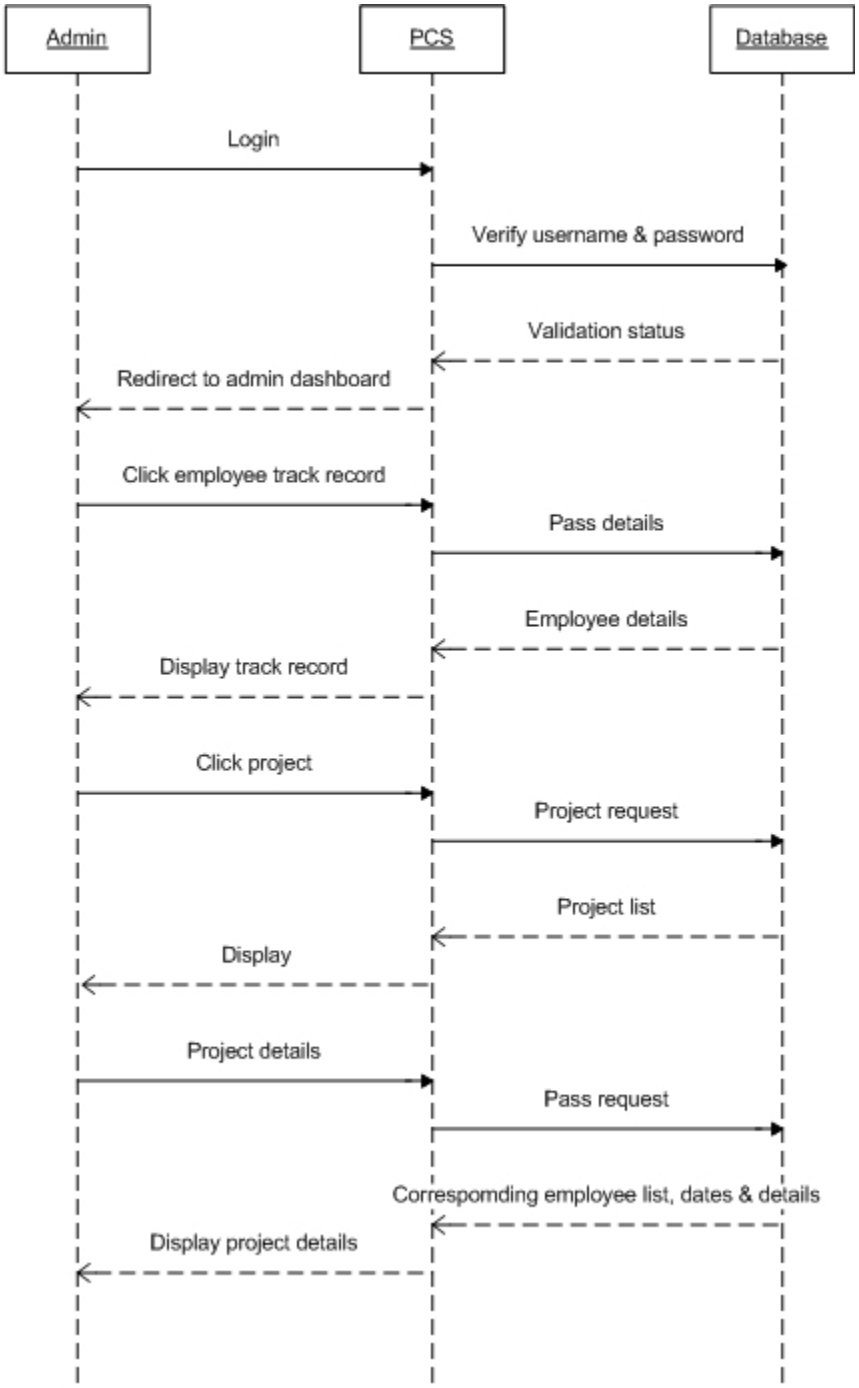
heads are synchronous calls, solid arrows with stick heads are asynchronous calls and dashed arrows with stick heads are return messages. This definition is true as of UML 2, considerably different from UML 1.x. Activation boxes, or method-call boxes, are opaque rectangles drawn on top of lifelines to represent that processes are being performed in response to the message (ExecutionSpecifications in UML). Objects calling methods on themselves use messages and add new activation boxes on top of any others to indicate a further level of processing. When an object is destroyed (removed from memory), an X is drawn on top of the lifeline, and the dashed line ceases to be drawn below it (this is not the case in the first example though). It should be the result of a message, either from the object itself, or another. A message sent from outside the

diagram can be represented by a message originating from a filled-in circle (*found message* in UML) or from a border of sequence diagram (*gate* in UML). UML 2 has introduced significant improvements to the capabilities of sequence diagrams. Most of these improvements are based on the idea of *interaction fragments* which represent smaller pieces of an enclosing interaction. Multiple interaction fragments are combined to create a variety of *combined fragments*, which are then used to model interactions that include parallelism, conditional branches, optional interactions, etc.

Sequence Diagram at User level:



Sequence Diagram at Admin level:



E-R Diagram

In software engineering, an **entity-relationship model** (**ERM**) is an abstract and conceptual representation of data. Entity-relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion. Diagrams created by this process are called **entity-relationship diagrams**, **ER diagrams**, or **ERDs**.

The first stage of information system design uses these models during the requirements analysis to describe information needs or the type of information that is to be stored in a database. The data modeling technique can be used to describe any ontology (i.e. an overview and classifications of used terms and their relationships) for a certain area of interest. In the case of the design of an information system that is based on a database, the conceptual data model is, at a later stage (usually called logical design), mapped to a logical data model, such as the relational model; this in turn is mapped to a physical model during physical design. Note that sometimes, both of these phases are referred to as "physical design".

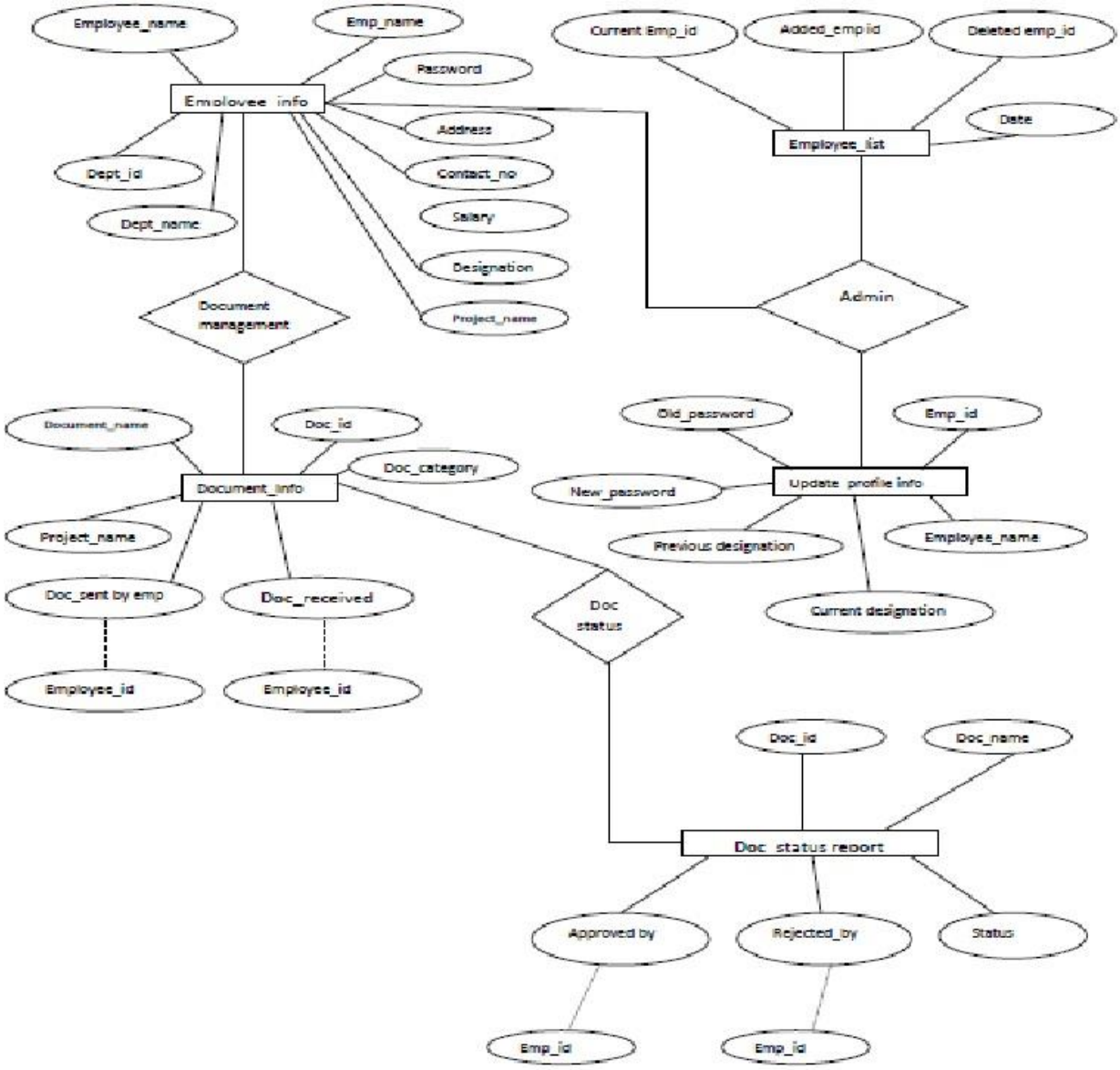
There are a number of conventions for entity-relationship diagrams (ERDs). The classical notation mainly relates to conceptual modeling. There are a range of notations employed in logical and physical database design, such as IDEF1X.

Primary key: An entity may be defined as a thing which is recognized as being capable of an independent existence and which can be uniquely identified. An entity is an abstraction from the complexities of some domain. When we speak of an entity we normally speak of some aspect of the real world which can be distinguished from other aspects of the real world. An entity may be a physical object such as a house or a car, an event such as a house sale or a car service, or a concept such as a customer transaction or order. Although the term entity is the one most commonly used, following Chen we should really distinguish between an entity and an entity-type. An entity-type is a category. An entity, strictly speaking, is an instance of a given entity-type. There are usually many instances of an entity-type. Because the term entity-type is somewhat cumbersome, most people tend to use the term entity as a synonym for this term.

Entities can be thought of as nouns. Examples: a computer, an employee, a song, a mathematical theorem.

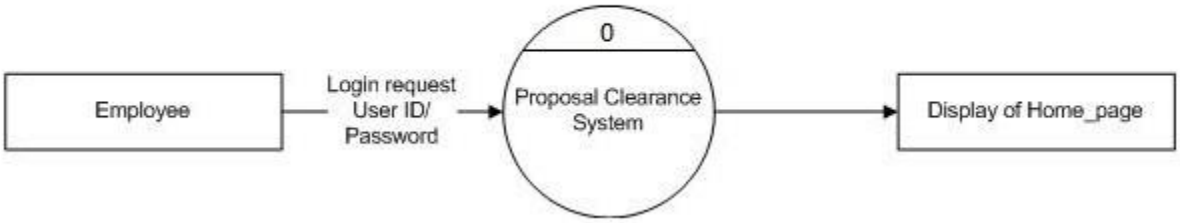
A relationship captures how two or more entities are related to one another. Relationships can be thought of as verbs, linking two or more nouns. Examples: an *owns* relationship between a

company and a computer, a *supervises* relationship between an employee and a department, a *performs* relationship between an artist and a song, a *proved* relationship between a mathematician and a theorem.

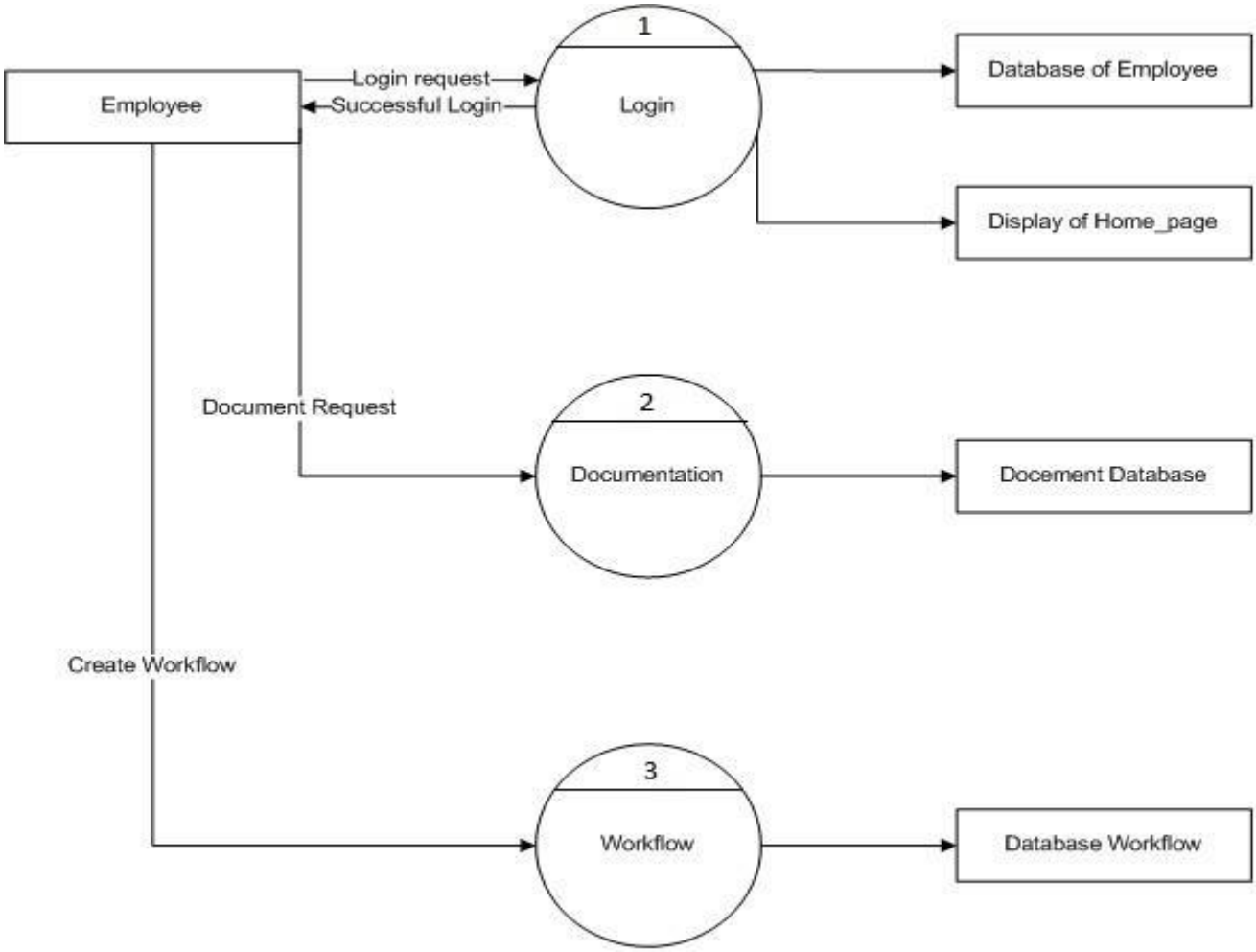


Data Flow Diadgrams:

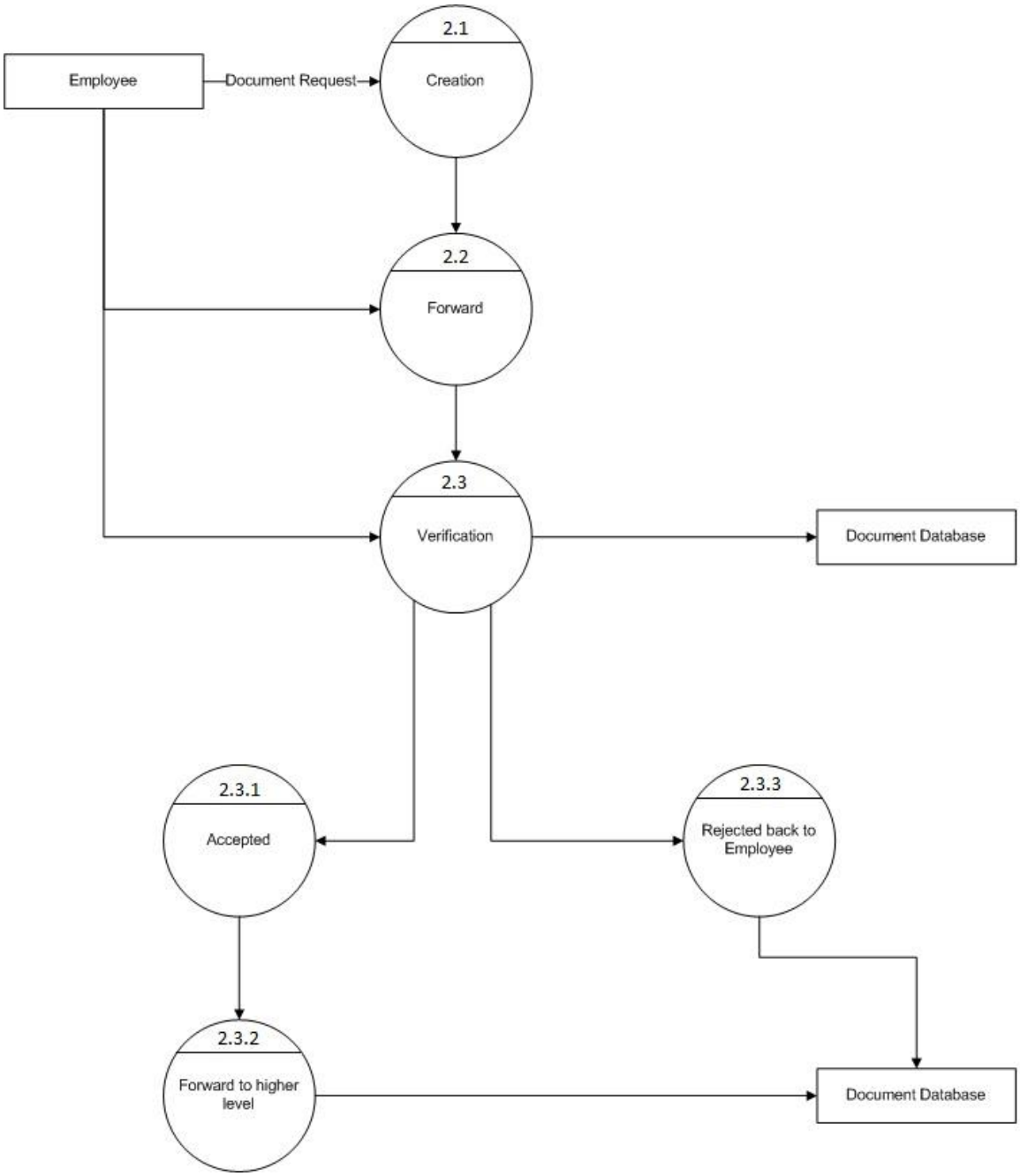
Level 0:



Level 1:

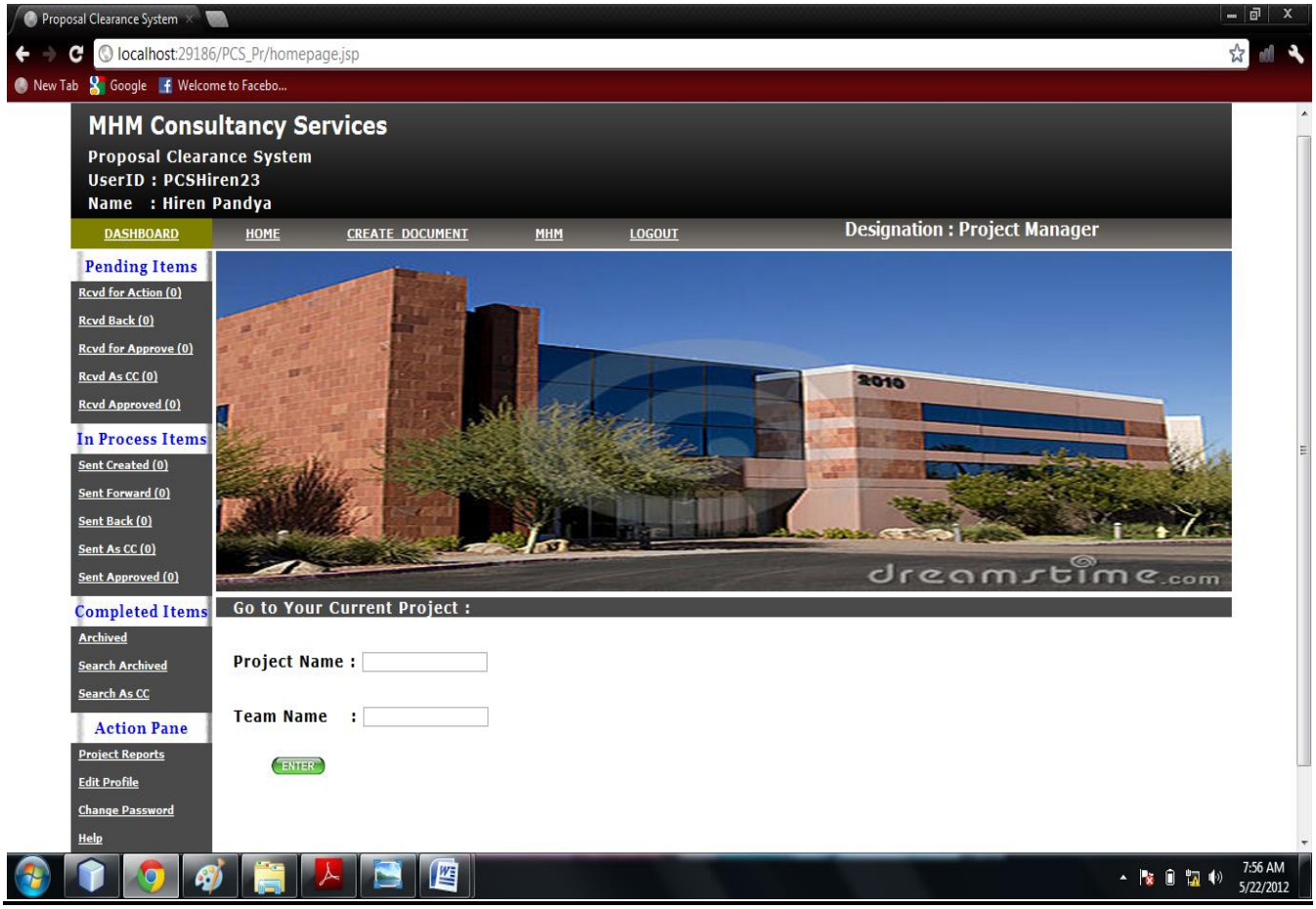


Level 2:

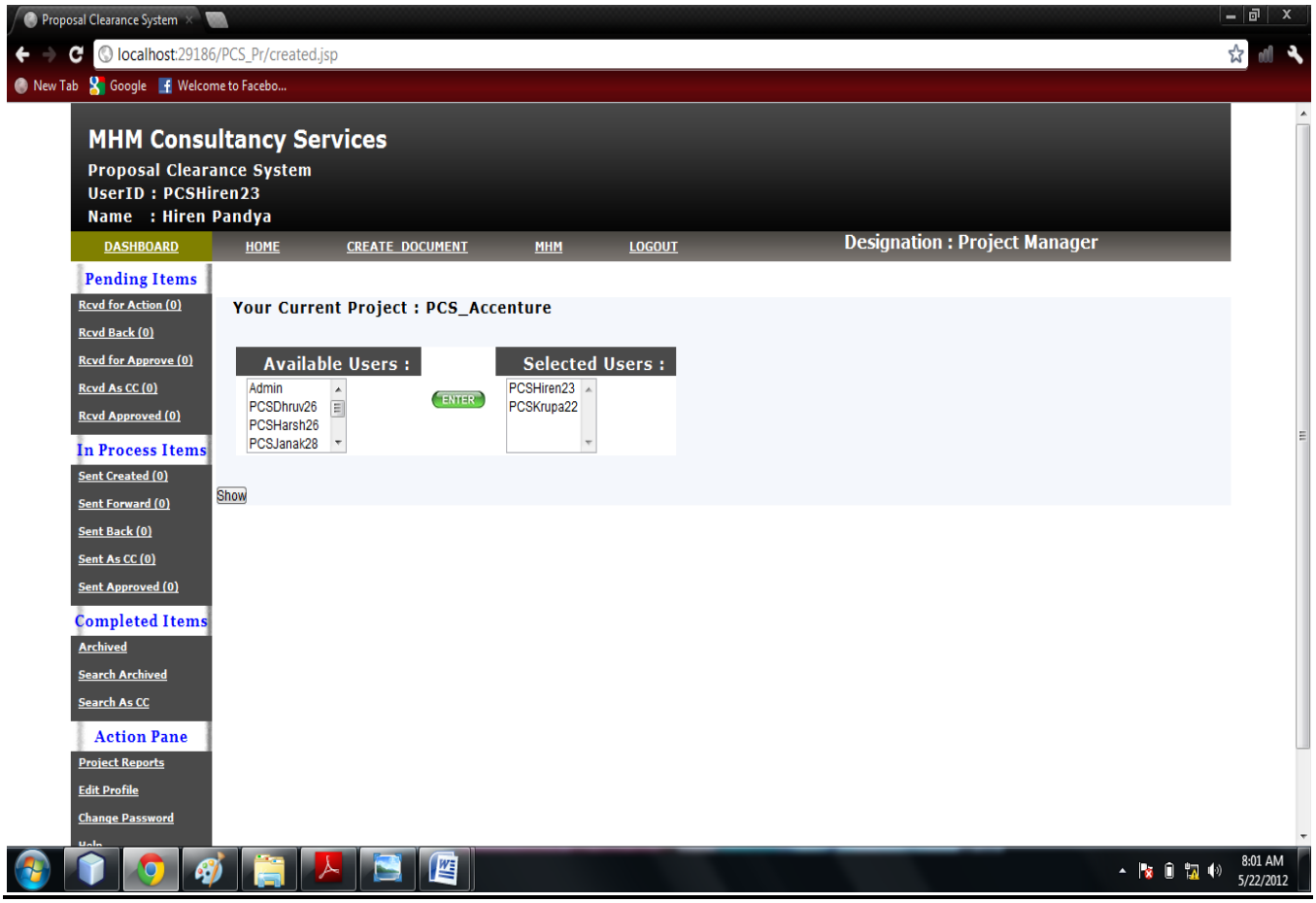

2.1
Employee
Document Request
Creation
2.2
Forward
2.3
Verification
Document Database
2.3.1
Accepted
2.3.3
Rejected back to Employee
2.3.2
Forward to higher level
Document Database

Important Screenshots:

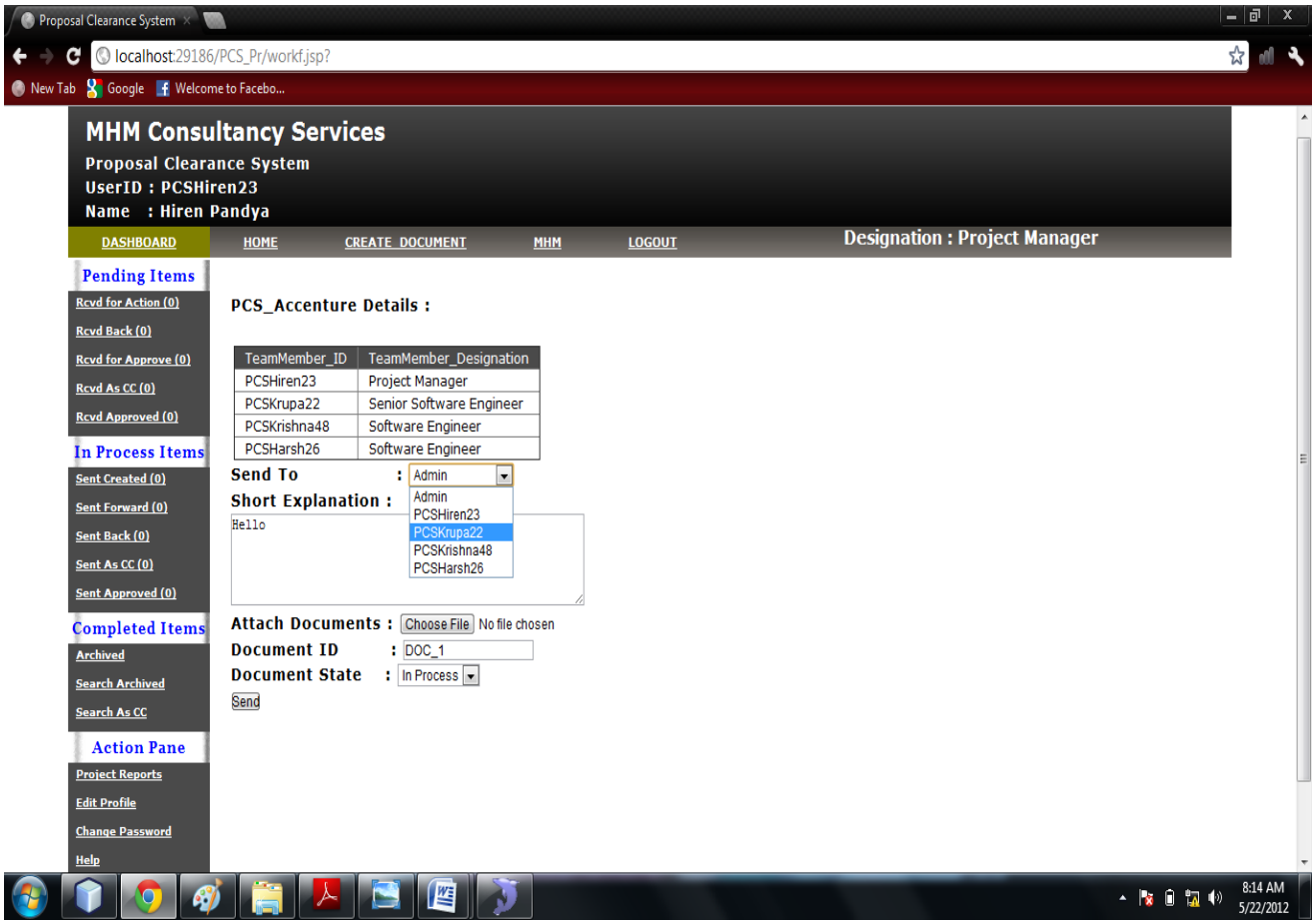
Home page:



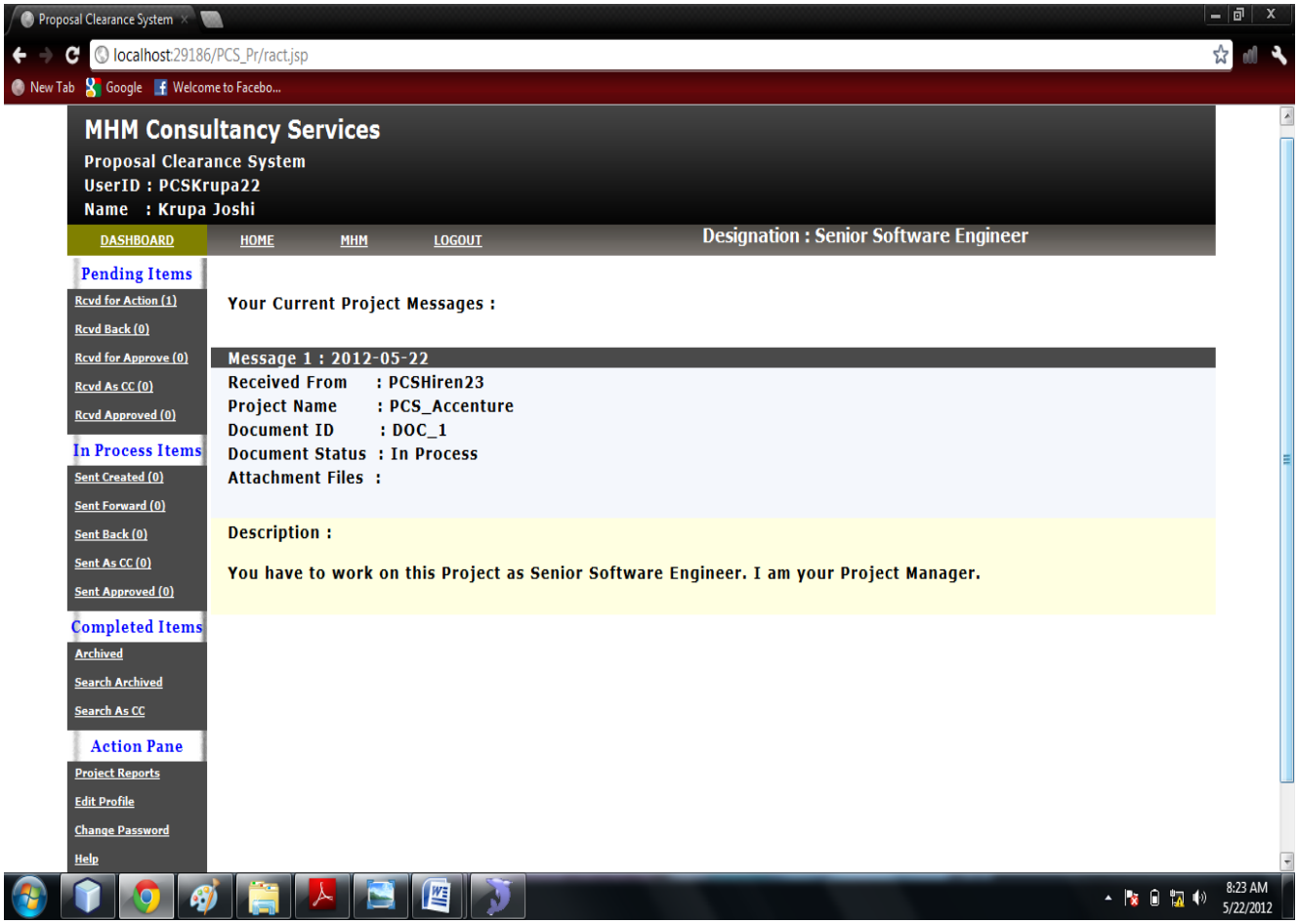
Documentation :



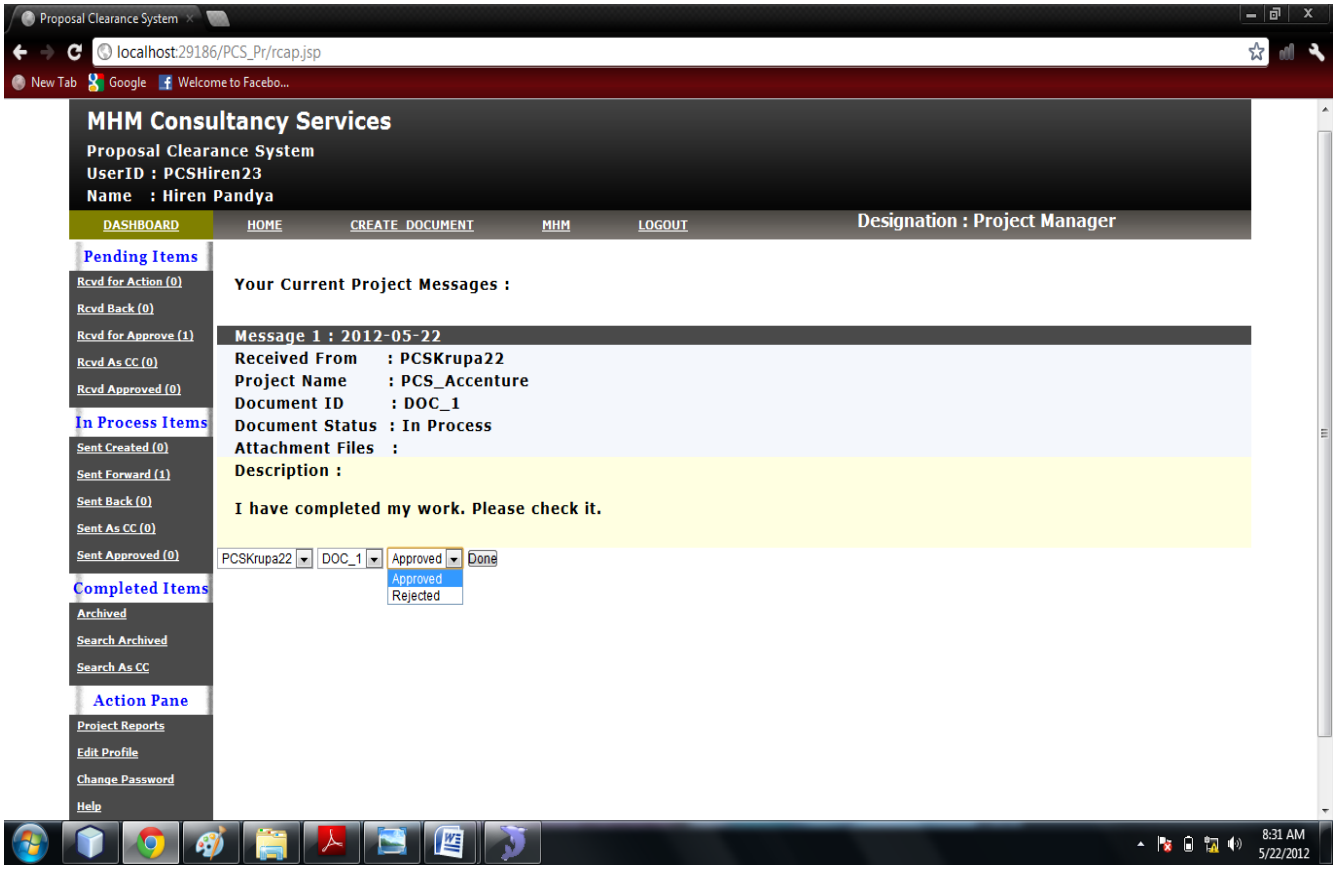
Document Attachment and Sending :



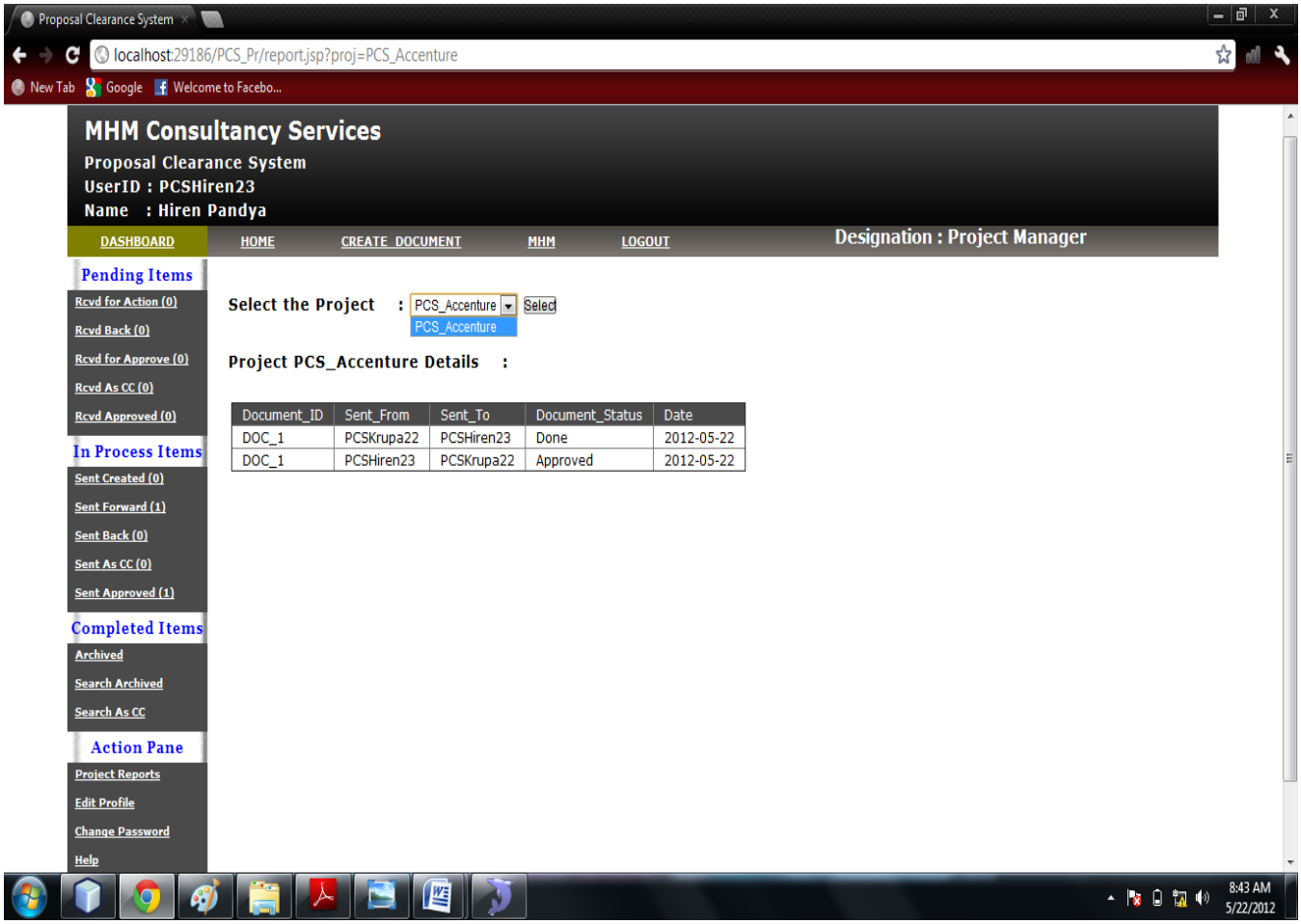
Received Message from higher entity:



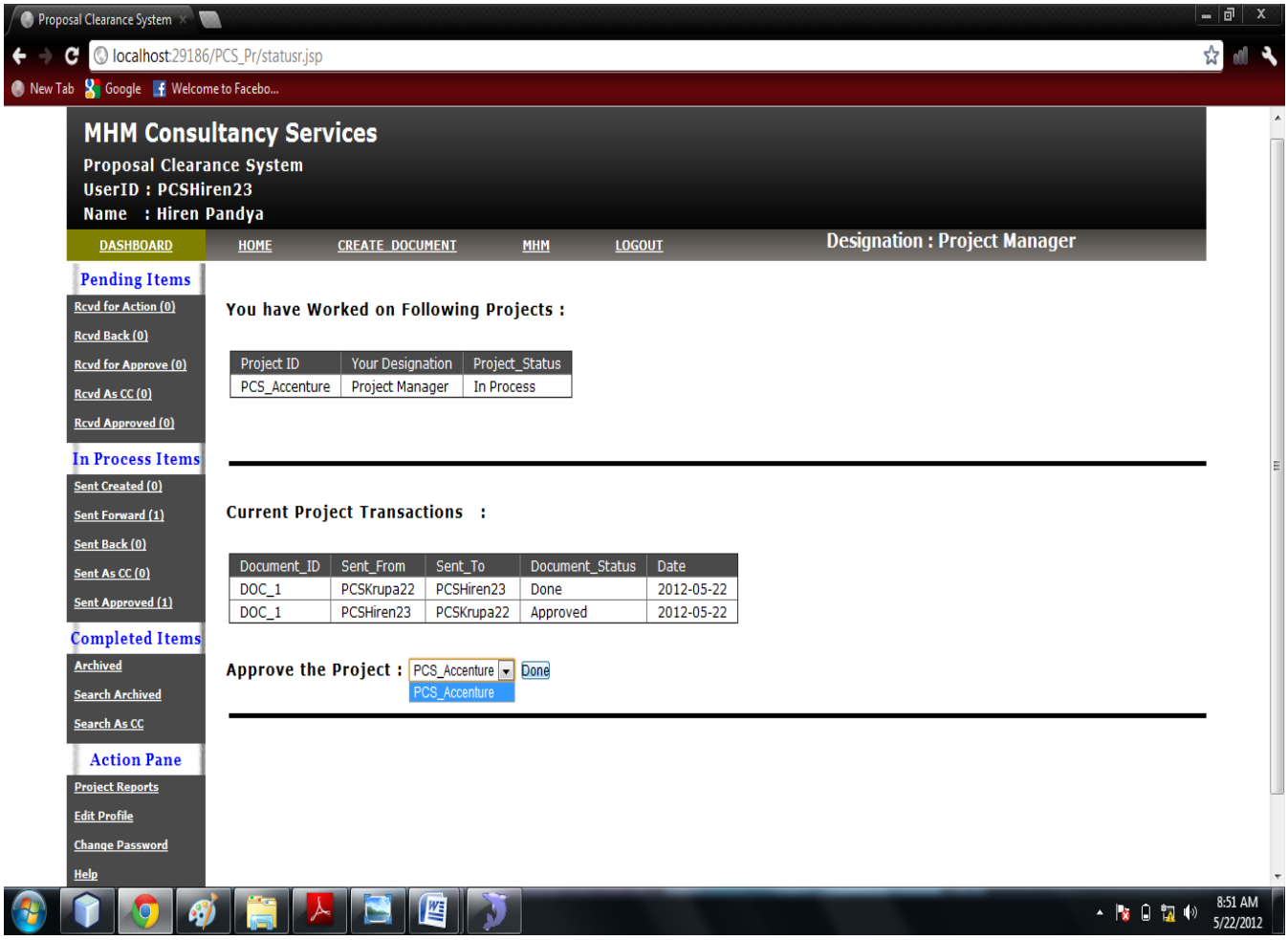
Document Approval from higher entity:



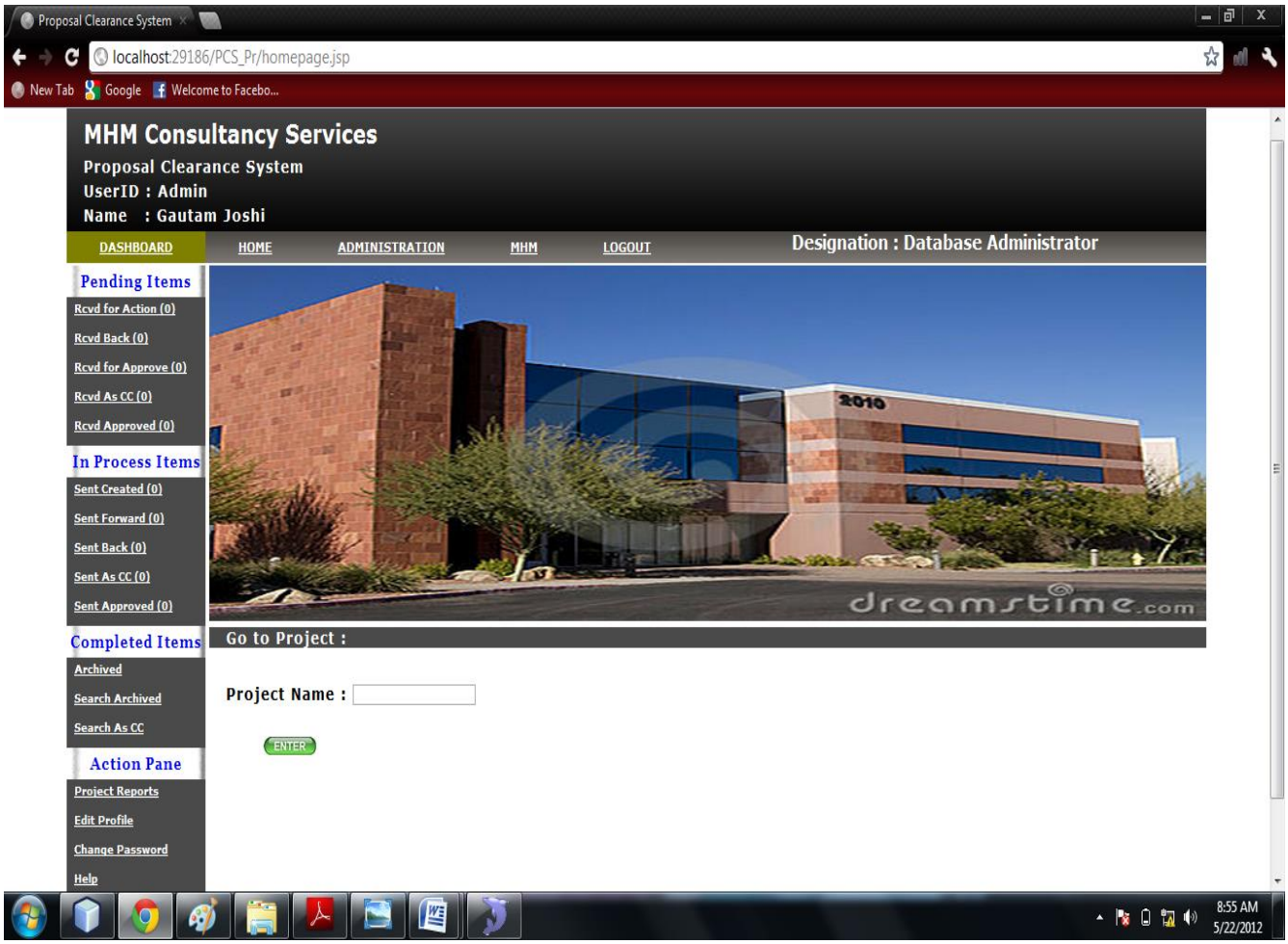
Transactions in selected Project :


Proposal Clearance System
localhost:29186/PCS_Pr/report.jsp?proj=PCS_Accenture
New Tab
Google
Welcome to Facebo...
MHM Consultancy Services
Proposal Clearance System
UserID : PCSHiren23
Name : Hiren Pandya
DASHBOARD
HOME
CREATE DOCUMENT
MHM
LOGOUT
Designation : Project Manager
Pending Items
Rcvd for Action (0)
Rcvd Back (0)
Rcvd for Approve (0)
Rcvd As CC (0)
Rcvd Approved (0)
In Process Items
Sent Created (0)
Sent Forward (1)
Sent Back (0)
Sent As CC (0)
Sent Approved (1)
Completed Items
Archived
Search Archived
Search As CC
Action Pane
Project Reports
Edit Profile
Change Password
Help
Select the Project : PCS_Accenture Select
PCS_Accenture
Project PCS_Accenture Details :
Document_ID Sent_From Sent_To Document_Status Date
DOC_1 PCSKrupa22 PCSHiren23 Done 2012-05-22
DOC_1 PCSHiren23 PCSKrupa22 Approved 2012-05-22
8:43 AM
5/22/2012

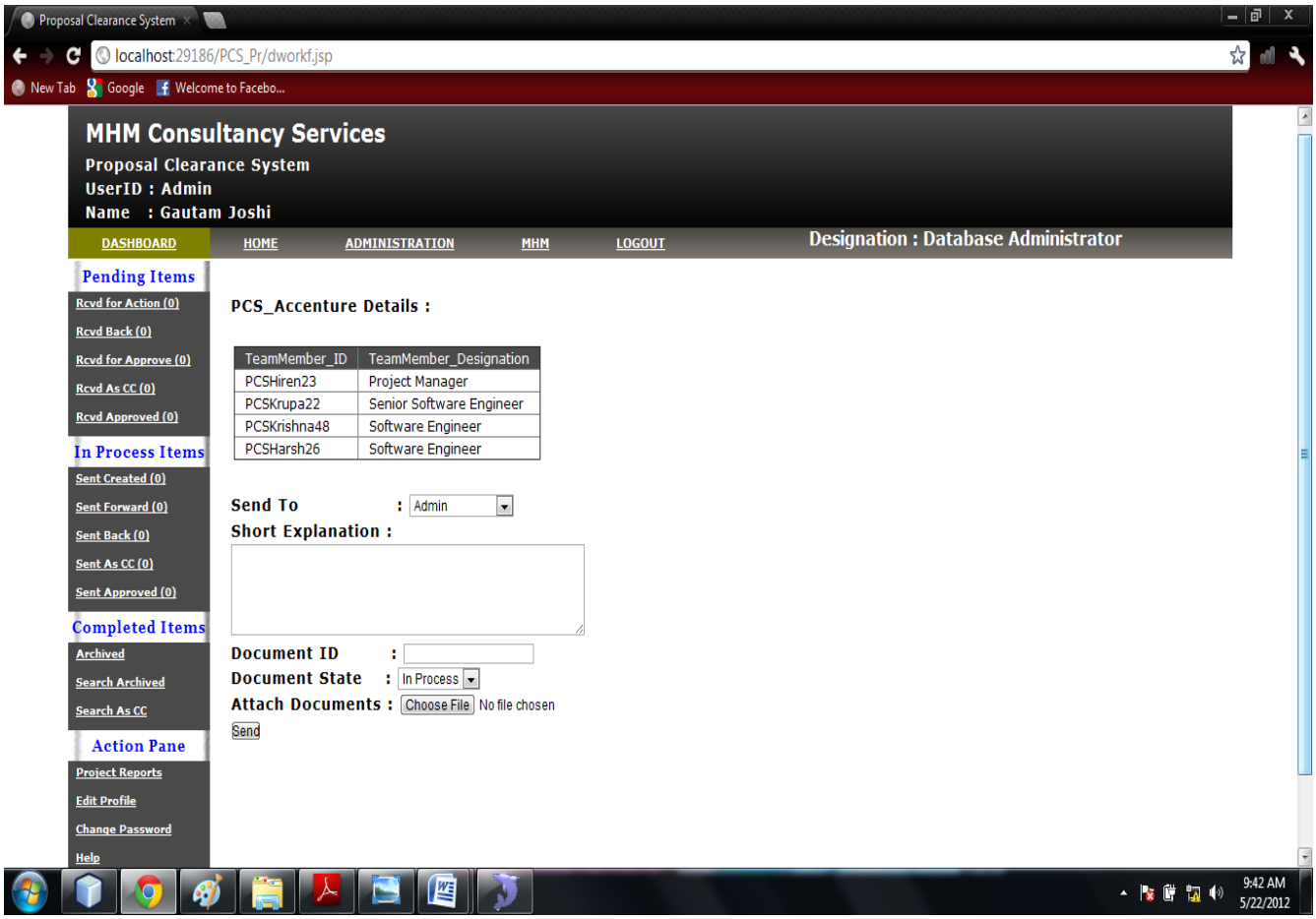
Transactions in Current Project and Project Approval functionality for top most entity :



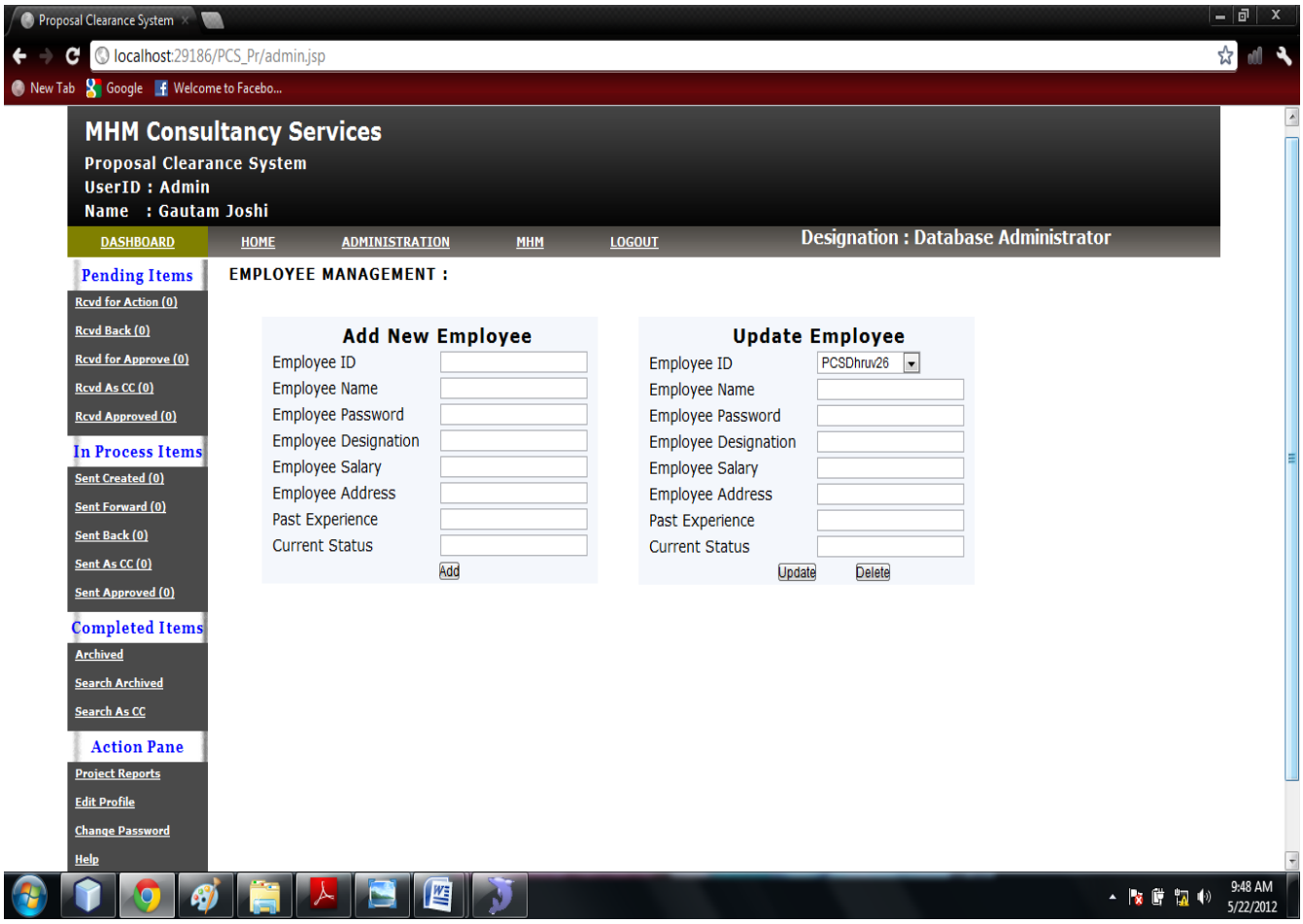
Login as Admin :



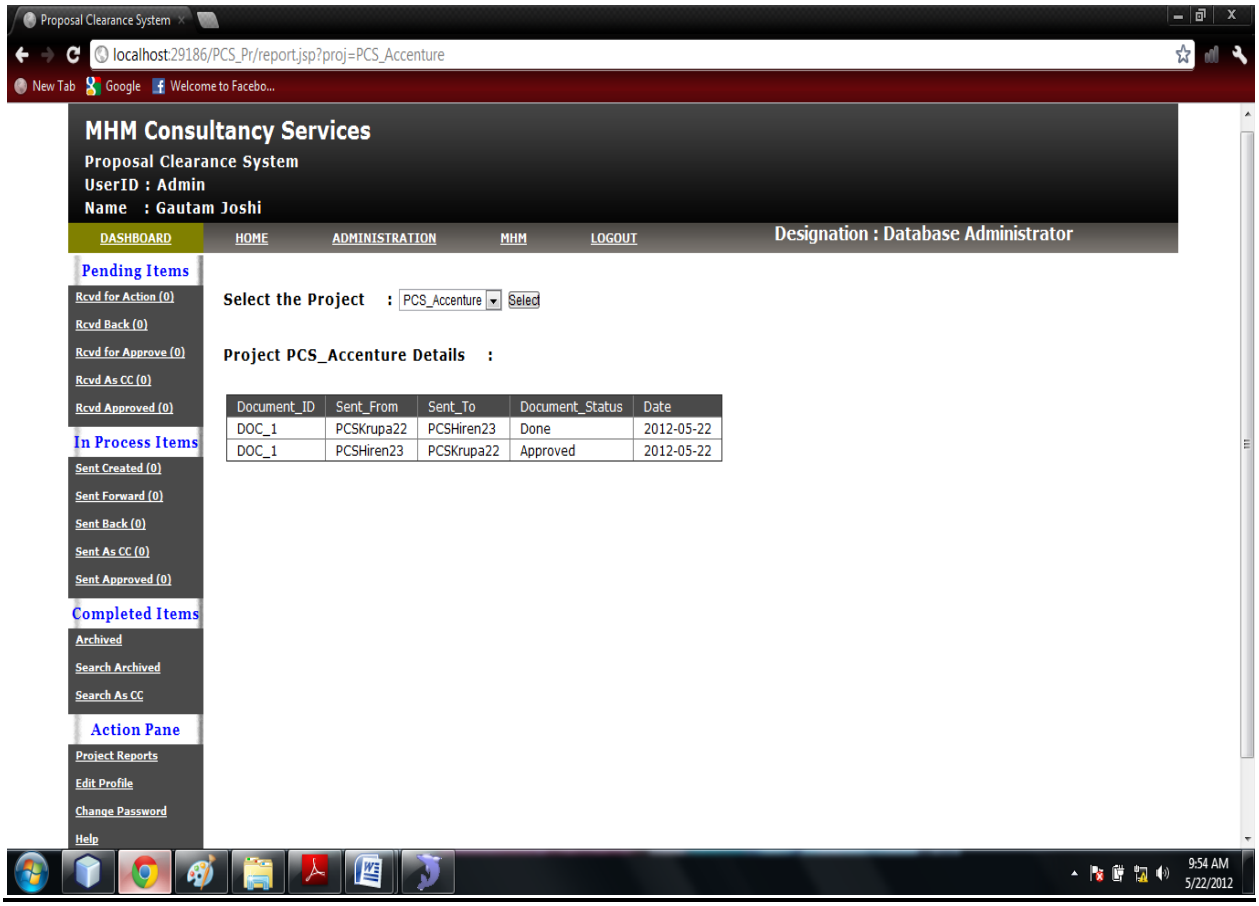
Admin can send message any employee for particular Project :



Admin can add, update and delete the employee :



Admin can see the Transactions of any particular Project :



5. Conclusion

The project definition came from the requirement of managing documents electronically and workflow management. The existing system for this is E-mail system, but the problem with this is, user is not notified when mail comes. Also companies have to allow employees to access E-mails n Internet. So, we are trying to develop the system which can solve all of this problem.

At first site, the project definition might look easy from the perspective of the number of modules or size. But it has a considerable level of difficulty in implementing it especially when we need to develop a system for a particular organization. Our focus of the application is the complete satisfaction of the user interacting with the application. Our application will be easy to understand and use.

REFERENCES